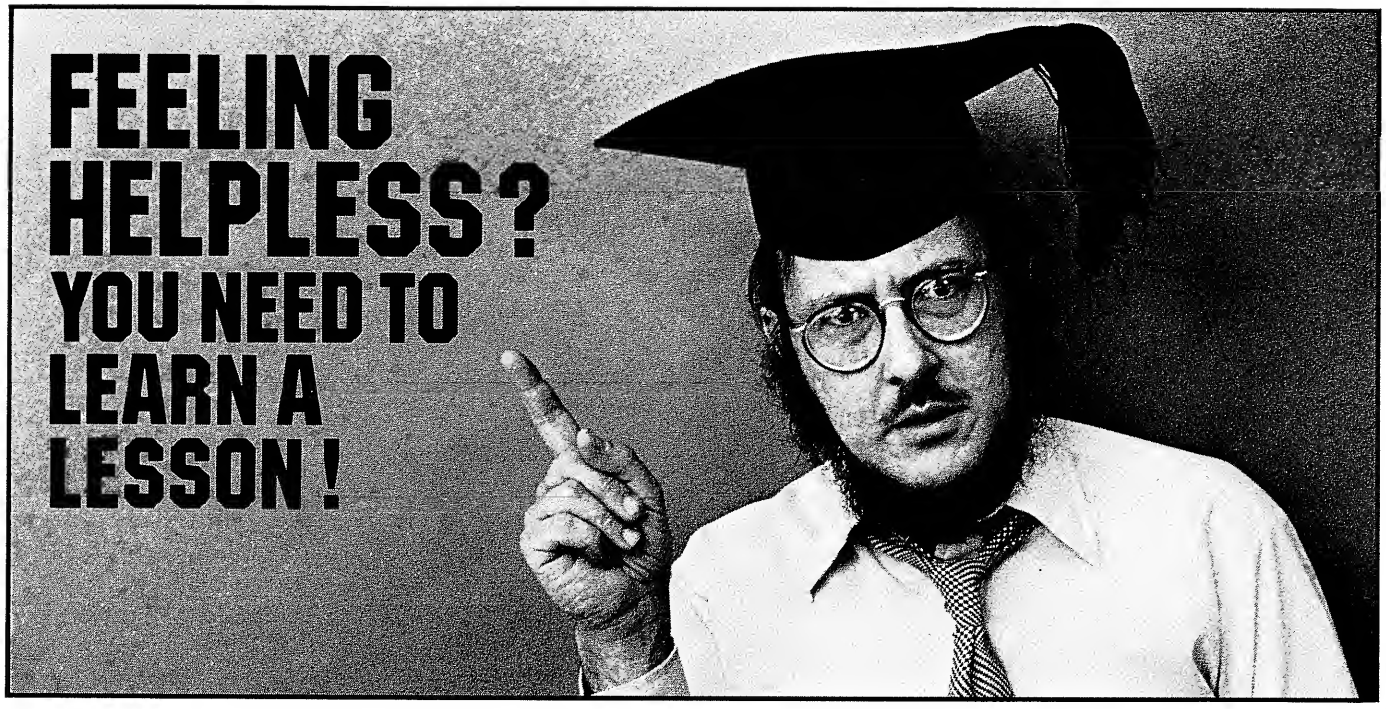
UNDERCOLOR

Vol. 1 No. 3 **January 11, 1985**



MICRO LANGUAGE LAB: LEARNING THE 6809

Feeling at the mercy of a programmer somewhere? Mystified by a machine? My **Micro Language Lab** will give you the power to end those helpless feelings.

Here's why. The heart of *any* computer program *anywhere* is machine language. Every piece of software uses it — your favorite game, spreadsheet, word processor, data base manager, or recipe file. Every time you hit "Enter", it's working. All the languages and operating systems are created from it. Basic, Fortran, Pascal, Flex, OS-9, TRSDOS... they're 6809 machine language, the whole lot!

And you can learn the language of the 6809, the programming heart of your Color Computer, with my **Micro Language Lab.**

Not everyone can program. Writing a few lines in Basic now and then doesn't make you a programmer. But if you *can* program, then my **Micro Language Lab** will teach you — the right stuff, the right way.

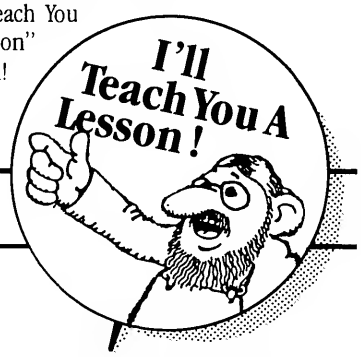
I'll teach you in 24 half-hour lessons on 12 audio cassettes, with a 220-page textbook, with data booklets, with 35 sample programs, and with a programming reference card. You'll spend 50 hours or more with my course, listening, watching, and working. And when you're done, you'll be programming your Color Computer in the 6809's language.

Stop ignoring those gnawing feelings of helplessness. The 6809 is where the power lies in your computer. Turn on that power with the **Micro Language Lab.**

- **Micro Language Lab,** $99.00 (plus $3.50 shipping and handling)

Requires 16K Extended Basic EDTASM +

Not sure? Write or call for a Table of Contents and sample pages — and your "I'll Teach You A Lesson" button!



ALSO AVAILABLE FROM GREEN MOUNTAIN MICRO

(Add $2.50 shipping and handling to your order)

Lowerkit III

- Full-time upper and lowercase installs in 15 minutes.
- Normal and reverse video standard
- Fully compatible with all Alpha and Graphic modes

$79.95 assembled and tested
$49.95 complete kit of parts
Important! Specify Color Computer or Color Computer II

TV Buff II, clear image for video monitors, $19.95 (specify CoCo or CoCo2)

CoCoPort parallel interface (now includes plastic case), $54.95 / $44.95 kit

ColorPack ROM/RAM pack, $29.95 (specify configuration) / $19.95 kit / case $3.00 extra

64K memory upgrade kit with full instructions, $49.95 / with memory test on tape, $54.95

Color Quaver Software Music Synthesizer on tape (requires 32/64K), $19.95

Color Burner EPROM Programmer (2716/32/32A/64/64A/128, 68764/66) with software, $69.95 / $56.95 kit

User Group, Educational, Club and Dealer Discounts are available.

TRS-80 and TRSDOS are trademarks of Tandy Corporation. Flex is a trademark of TSC, Inc. OS-9 is a trademark of Microware Corporation.

Green Mountain Micro
Bathory Road, Box U
Roxbury, Vermont 05669
802 485-6112
Hours: 9am - 5pm EST, Monday - Friday
COD/VISA/MASTERCARD

Who's Who

Editor
Debra A. Marshall

Technical
Dennis Bathory Kitsz

Contributing Photographer
Charley Freiberg

Production Manager
Clare McCarthy

Paste-Up
Janet Patterson

Logistics Coordinator
Joanne Trottier

Technical Consultant and Publisher
Dennis Kitsz

Advertising:
Judy Knapp
(802)485-6139

Advertising:
Spencer Knowlton
(207)785-4151

Office
(802)485-6440

Editorial Office
(207)785-5148

Published By: ColorPlus Inc., Box 6809, Roxbury, VT 05669. (802)485-6440.

Circulation Problems? Contact David Gunn, Box 6809, Roxbury, VT 05669, 802-485-6440.
Subscriptions: Until Dec. 1, 1984: $20. After Dec. 1: $33. Add $14 for first-class postage. Foreign: until Dec. 1, 1984: Canada/Mexico $30 U.S.; other countries $40 U.S. *Under Color* is published 26 times a year. Call 802-485-6440, or write to *Under Color,* Box 6809, Roxbury, VT 05669, Attn: Perley Swann.
Advertisers: rates available.
Editorial Office: c/o Debra Marshall, RFD 1, Box 786, Union, ME 04862 207-785-5148. Submissions are welcome; include S.A.S.E. for returns. To receive author's guide, write: Author's Guide, Box 6809, Roxbury, VT 05669. Letters to Editor, manuscripts, article queries, etc. should be sent to editorial office.

Table of Contents

• **Another Fine Mess** ..... page 2
By Richard Ramella — *Classy circle art.*

• **The Data Gatherer—Pt. I** . page 8
By Dennis Kitsz — *A welcome interruption.*

• **Cross Talk—Pt. II** ..... page 12
By William Barden, Jr. — *The last word on RS-232*

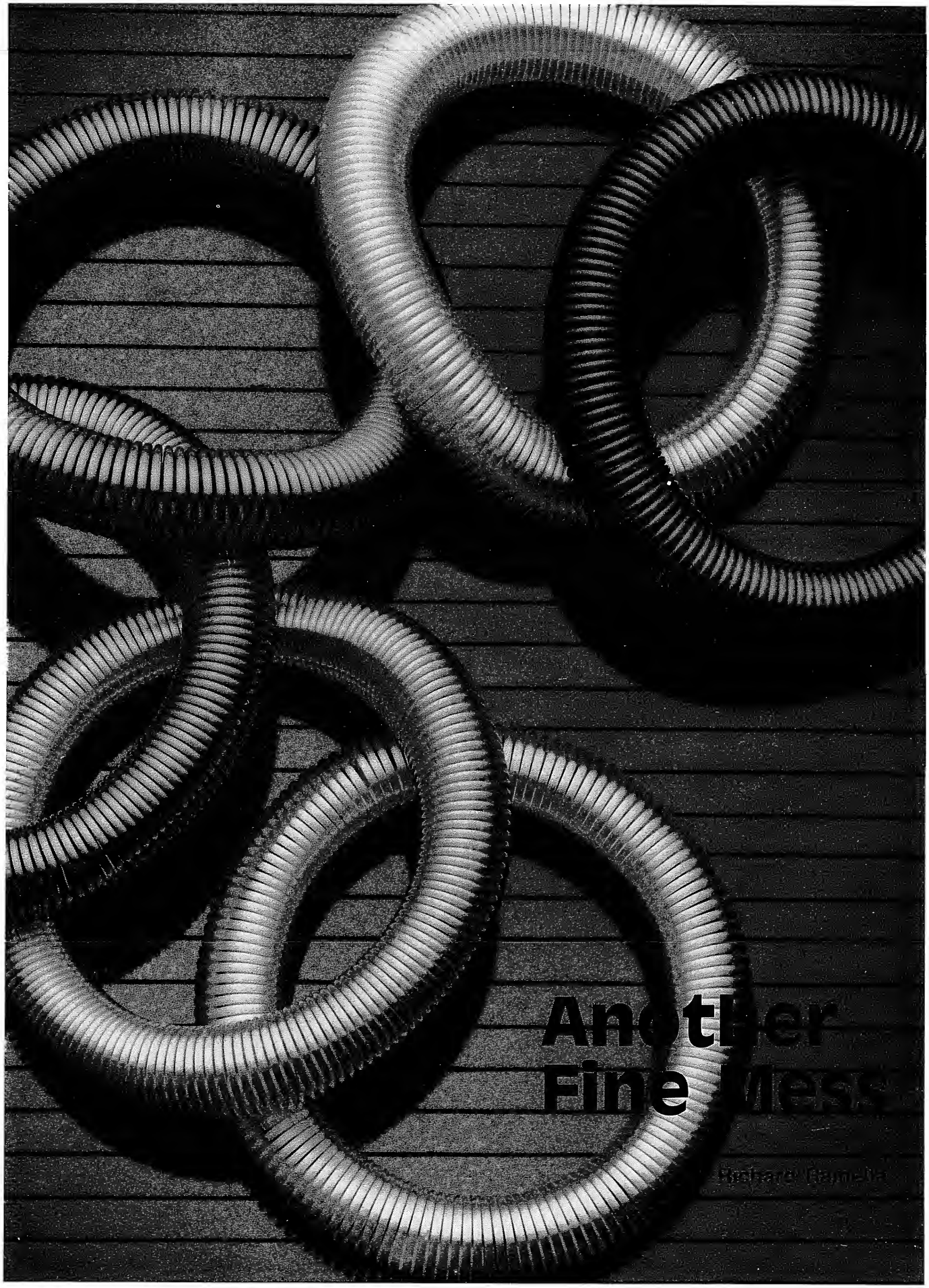
• **Find** ..... page 15
By Stephen Allen — *A new word in Basic.*

• **News and Gossip** .....page 21
By Dennis Kitsz — *Hot off the modem.*

• **Reviews** ..... page 22

• **Defuser** ..... page 24
By Terry Kepner — *Answers to life's little problems.*

Cover Photo by Charles Freiberg


Another
Fine Mess

Ed's Note: Here's some fun just in time for the end of the holidays, as only Richard Ramella can product it!

With the help of Leonardo da Vinci, Stanley Laurel, Oliver Hardy and Winky the Clown, I'm going to show you how to use the Circle command to create simple and impressive computer art.

Normally computer art tends to produce lines and squares, and complex efforts require more data entry than the results often merit. However, with the Circle command's diversity you can break into the heady atmosphere of the curved line. You can control the curve and bend it to your artistic will.

I must acknowledge da Vinci, perhaps the greatest artist in history, because of an anecdote I read about him. It is said he could draw a perfect circle with an unsupported brush. Out he would reach, around would go his sure hand and–swish—a perfect circle.

I began to look for the circle in Leonardo's art. It is everywhere. You probably won't have much trouble finding a print of the *Mona Lisa.* Look carefully at that famous face. The forehead, the cheeks, the chin, the nose, even the enigmatic smile—they are all parts of circles. Somehow, the harmony and the mystery of the face emerge from the gently rounded parts.

Perhaps it was a leap from the sublime to the mundane, but I began to see that the Circle command could be used to make art. I'm no artist, but I tried it and it worked. You won't see anything comparable to *The Last Supper* in my work. Try Listing 1, *Another Fine Mess*, and you will see passable caricatures of my favorite comics, Mister Laurel and Mister Hardy. Try Listing 2, *Winky*, and you will see a colorful clown with a bit of animation.

What is noteworthy about these efforts is that the Circle command is so diverse it can be used to create greatly different curved lines.

Listing 3, *Circle Tester*, is a demonstration program. It lets you play with three concepts taught in the *Going in Circles* chapter (pages 40–47) of the manual *Going Ahead With Extended Color Basic*. The three ideas of the demonstration are radius of the circle, height/width ratio, and start-end points of the circumference.

I urge you to key in and try the short program *Circle Tester* before you try *Winky* or *Another Fine Mess.* This way, you'll understand the concepts of the main programs when you

Listing 1. Another Fine Mess

```
100 REM * ANOTHER FINE MESS * RICHARD RAMELLA
110 REM * TRS-80 EXTENDED  COLOR BASIC 16K
120 CLS
130 CLEAR 400
140 PRINT @ 231,"ANOTHER FINE MESS"
150 FOR T=1 TO 1000
160 NEXT T
170 PMODE 4,1
180 PCLS 1
190 SCREEN 1,0
200 FOR T=1 TO 500
210 NEXT T
220 CIRCLE(50,96),35,.2,3
230 X=50
240 FOR Y=25 TO 30
250 CIRCLE(X,Y),40,.2,.1,.01,.5
260 NEXT
270 PAINT(40,2),2,2
280 FOR Y=135 TO 136
290 CIRCLE(50,Y),20,4,5,0,.5
300 NEXT Y
310 FOR Y=109 TO 111
320 CIRCLE(50,Y),20,2,.2,0,.5
330 NEXT Y
340 Y=70
350 FOR X=35 TO 65 STEP 30
360 CIRCLE(X,Y),10,2,3,.5,0
370 CIRCLE(X+3,Y-3),5,2
380 PAINT(X+3,Y-3),2,2
390 NEXT
400 LINE(55,75)-(68,100),PRESET
410 LINE(45,75)-(54,100),PRESET
420 CIRCLE(61,100),8,2,.2,0,.5
430 FOR X=15 TO 85 STEP 70
440 CIRCLE(X,70),4,2,5
450 CIRCLE(170,96),70,2
460 FOR Y=155 TO 156
470 CIRCLE(170,Y),20,2,.25,.5
480 NEXT Y
490 CIRCLE(170,110),15,2
500 FOR A=12 TO 1 STEP -1
510 CIRCLE(170,147),A,.2,2,.5
520 NEXT A
530 FOR X=145 TO 185 STEP 40
540 CIRCLE(X,90),5,2,.8,.25,.75
550 CIRCLE(X+5,90),3,2
560 NEXT X
570 DRAW "C0;BM100,50;R140"
580 CIRCLE(170,50),55,2,.8,.5,.01
590 DRAW "C1;BM170,22;D5"
600 PAINT(170,30),2,2
610 FOR X=140 TO 200 STEP 5
620 CIRCLE(X,57),5,2,2,.25,.75
630 NEXT X
640 POKE 223,100
650 A$="L8;G;P64;E;P64;G;P64;E;P16"
660 B$="L10;FGFDG;L8;E;P4"
670 C$="L12;FGFDG;L8;C;P4"
680 D$="L8;E;L4;C;P8"
690 E$=D$+D$+D$
700 F$=A$+A$+B$++A$+A$+C$+E$
710 PLAY F$
720 GOTO 710
```

see them at work.

Circle Tester has a three-item menu. Each one either draws a circle or a portion of a circle. In all cases the center of the circle is the center of the screen: 128 on the across axis and 96 on the down axis.

Let's try each menu item:

Radius—Run the program and tap R. The radius of a circle is the distance from its center to its edge. On the Color Computer the numbers referring to the radius are expressed in terms of the PSET points on the screen at the highest resolution. That means the screen has 255 points on the X axis, and 192 on the Y axis.

For some strange reason we are allowed a circle radius from 0 to 65535. In trying this test, you will see the following prompt:

```
FOR RADIUS OF CIRCLE
- 0 to 65535
```

Answer the prompt with a number within these two figures and tap Enter. When you have seen the result and want to try again, tap any key to return to the prompt.

Line 640 shows the form of the command, and the X position is where the radius number or variable goes in the circle command.

In choosing radii, the number 0 will produce a mere dot on the screen. Try larger numbers—5, 10, 30 and 40—to see larger circles. As you choose increasingly larger numbers for the radius, the circle, like a balloon inflated in a box, will flatten against the edges of the screen. Keep trying higher numbers and you will in time see the result as a square at screen's edge.

Now run the program again. Type H and tap Enter to try some height/width experiments. Depending on the number you enter here, you can make either height or width greater than the other. You will see this prompt:

```
— 1 TO 255 FOR HEIGHT
— 0 TO .99 FOR WIDTH
```

Answer the prompt by typing the number and tapping Enter. To return from the display to the prompt for another try, tap any key. Line 300 is where the Circle command does its trick, and the variable X at the end of the command represents

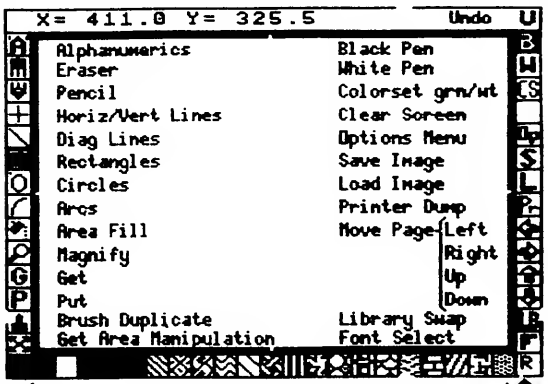
Listing 2. Winky

```
100 REM * WINKY THE CLOWN * RICHARD RAMELLA
110 REM * TRS-80 EXTENDED COLOR BASIC
120 PMODE 3,1
130 PCLS 1
140 SCREEN 1,1
150 CIRCLE(128,100),25
160 CIRCLE(96,70),15
170 CIRCLE(158,70),15
180 LINE(0,191)-(78,160),PSET
190 LINE(178,160)-(252,191),PSET
200 CIRCLE(96,70),10,3,.70
210 PAINT(97,71),3
220 CIRCLE(158,70),10,3,.7
230 PAINT(157,71),3
240 CIRCLE(96,70),3,,2
250 CIRCLE(158,70),3,,2
260 PAINT(130,96),8,8
270 CIRCLE(128,96),60,4,2
280 CIRCLE(60,96),10,,3
290 CIRCLE(196,96),10,,3
300 CIRCLE(100,60),20,,2,.5,1
310 CIRCLE(154,60),20,,2,.5,1
320 DRAW "BM30,30;R50"
330 DRAW "BM178,30;R50"
340 LINE(30,30)-(70,70),PSET
350 LINE(228,30)-(188,70),PSET
360 PAINT(40,35),3,8
370 PAINT(180,35),3,8
380 PAINT(10,191),8,8
390 PAINT(250,191),8,8
400 CIRCLE(128,96),200,4
410 Y=2
420 FOR X=96 TO 160 STEP 4
430 LINE(X,0)-(X,Y),PSET
440 IF X<127 THEN Y=Y+4 ELSE Y=Y-4
450 NEXT X
460 DRAW "BM128,127;D11"
470 DRAW "BM110,145;R35"
480 PAINT(3,3),2,4
490 PAINT(250,3),2,4
500 PMODE 4,1
510 FOR T=1 TO 800
520 NEXT T
530 DRAW "C0;BM100,145;R55"
540 FOR X=125 TO 155
550 IF X=126 THEN FOR T=1 TO 1000: NEXT
560 CIRCLE(128,X),40,1,.5,0,.5
570 NEXT X
580 FOR X=145 TO 151
590 CIRCLE(128,X),32,2,.4,0,.5
600 NEXT X
610 FOR T=1 TO 1000
620 NEXT T
630 PMODE 3,1
640 SCREEN 2,1
650 FOR T=1 TO 500
660 NEXT T
670 FOR Q=1 TO 2
680 FOR X=10 TO 3 STEP -1
690 CIRCLE(96,70),X,5,.7
700 CIRCLE(96,70),3,,2
710 NEXT X
720 FOR X=1 TO 10
730 CIRCLE(96,70),X,3,.7
740 CIRCLE(96,70),3,,2
750 NEXT X,Q
760 GOTO 670
```

FULL GRAPHIC CAPABILITIES !!

VIZIDRAW 1.0

BY GEORGE ZIRIENICZ AUTHOR OF COLOR TOOLKIT



Interactive Drawing Board

Graphic Display Processor

Text Manipulation

Versatile Text Sizes/Fonts

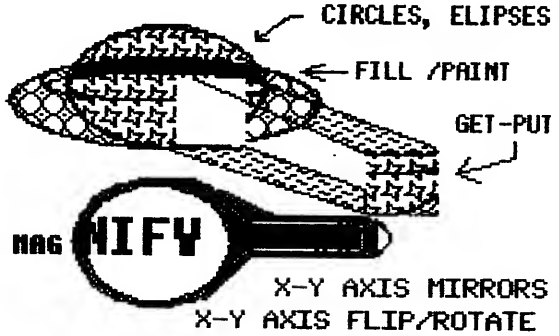
Italics Underline **Bold**

Small, Normal Size or **ENLARGED !**

448 X 480 PIXEL WORKSHEET !
You can design a **whole sheet** of paper in the COCO's memory!
NO MORE SMALL PICTURES!!

THE POWER OF 'MACPAINT' AT A COCO PRICE !

13 STANDARD FONT SETS INCLUDED !!
(3x5, 4x7, 5x7, 7x9, Daisy, Venice, English, Frontier, Hollow, Future, Digital 1 and Digital 2)



X-Y AXIS MIRRORS
X-Y AXIS FLIP/ROTATE
'PUT' ACTION/TILT

PRINTERS: RS (LP VII FORMAT, EPSON, C-ITOH 8510, GEMINI 10 & 10X)

TAPE or DISK please specify **$49.95** + $2.00 POSTAGE

(DEALER INQUIRIES WELCOME)

VIDX VIDEO DIGITIZER

NO DISK DRIVES NEEDED !!
NO Y-CABLES or MULTIPACKS!

Yes, that's right you don't need any of the above to run VIDX. VIDX can be run with a simple 16K E.C.B. TAPE MACHINE up to a 64K DISK SYSTEM!!

Just connect VIDX between the video source and the RS-232 port of your Color Computer. You can digitize any source of video such as off air TV, video tape recorder or live camera! Software is supplied on a RS-DOS DISK or cassette (not protected!).

Please specify system you plan to use

THE SOFTWARE FEATURES:

-] SINGLE/ MULTIPLE EXPOSURES
-] ABORT OR RESUME SCAN
-] LOGICAL 'AND' SCREEN 2 AND 1
-] FRAME PICTURE
-] READ/WRITE TO GRAPHICOM, RS-DOS DISK OR CASSETTE

VIDX-1 - $99.95 DIG.&SOFT.
VIDX-2 -$124.95 DIG.&SOFT.
W/ 2 PORT RS-232 SWITCHER
PLUS $3.00 POSTAGE

GRAFX-MONTHLY GRAPHICS DISK

A MONTHLY COLLECTION OF ARTS SEASONAL MOTIFS AND HUMOR WITH MUSIC! JUST INSERT INTO DRIVE!

SPECIAL NEW PRICE $75.00
YEAR'S SUBSCRIPTION
3 MONTH TRIAL ONLY $25.00
SAMPLE DISK $10.00

TO ORDER SEND YOUR CHECK OR M.O. TO:

G R A F X
P. O. B O X 2 5 4
WEST MIFFLIN, PA 15122-0254

VOICE LINE (412) 466-6974 (6-9PM)
24 HR BBS- COCONET (412) 744-2335- 300/1200BAUD
- YOU CAN DOWNLOAD DIGITIZED PIX -

THIS COMPLETE AD PREPARED BY VIZIDRAW WITH NO CUTS OR PASTES!--COLOR COMPUTER TM TANDY CORP MACPAINT TM APPLE COMPUTERS

the number you've chosen. The number 1 is meant to produce a perfect circle. The number 2 gives a circle which is twice as high as it is wide, and 3 gives a 3-to-1 height/width ratio. Larger numbers, such as 255, squeeze the top and bottom of the circle against the screen, producing a box. Experiment with varied numbers to get the idea here.

By entering a number from 0 to .99, the height is made that percentage of the width. Example: the number .5 would give a circle half the height of its width. Again, experiment.

The final menu choice of Circle Tester is Start–End. It is possible to use the circle command to draw a section of a circle (an arc) which begins and ends at points you specify. This method is explained on page 45 of the manual.

In trying Start–End, you must enter two numbers separated by a comma. The first is the start point, the second the end point. Line 380 of the program shows the form with X representing start and Y the end point.

As in other menu choices, type your number choices and tap Enter to see the result. Then tap any key to return to the menu. Try these possibilities: .5,.75 / .25,.35 / 0,.80 and see what happens.

You might want to later go through the manual's chapter on circles. I have left out a few possibilities. I wanted only to cover the major concepts used in the main programs with this article.

When you have tried *Circle Tester*, I hope the circles and curved lines produced will have suggested possibilities to you. You now have the capability of producing an amazing variety of shapes. In the programs ahead you will see eyes, ears, noses, smiles, hair, heads . . . all produced with different kinds of circles.

Now it's time to try *Another Fine Mess*. It produces caricatures of Laurel and Hardy and plays a version of the duo's theme, which is known as the *Cuckoo Song*.

Let me cover the program:

• Laurel Sequence

220 Head shape.

230–260 Hat brim. Across axis of circle's center is X, which equals 50. By looping Y from 25 to 30, the lower part of a circle is drawn several times and thickens into the brim.

270 The rest of the hat is painted in.

280–300 Chin created and given a bit of emphasis by forming two lower circle sections with Y having subsequent values of 135 and 136.

310–330 Wide bottom half of circle drawn with Y having three descending positions, and smiling mouth results.

340–400 Eyes created. 310 begins a loop that occurs twice 30 positions apart. In each pass . . . 360 forms the large arch of the eye, 370 draws the circle of the eye's iris, 380 paints in the eye, 390 sets a light dot in the eye to give it sparkle, and 400 is the end of the loop.

410–420 These two line commands form the long ridges of the nose.

430 The rounded bottom of the nose connects the nose's two ridge lines.

440–460 Ears are made on either side of the head.

• Hardy Sequence

470 Head Shape.

480–500 A loop draws the mouth in a scowl of two thicknesses.

510 The bulbous nose

520–3540 Mustache is created by forming a high, elongated half circle, then stepping it down from the radius of 12 to 1.

560–580 Eyes. They are drawn in a loop which centers one eye on an X value on 145, the other on an X value of 185. Line 560 forms the found area to the left of the eye, and line 570 forms the iris.

590 The hat brim

600 The rounded derby's shape.

610 A small line is drawn, linking the top and bottom areas of the hat. This makes it possible for the coming single Paint command to spill over entirely within the derby.

continued on page 23

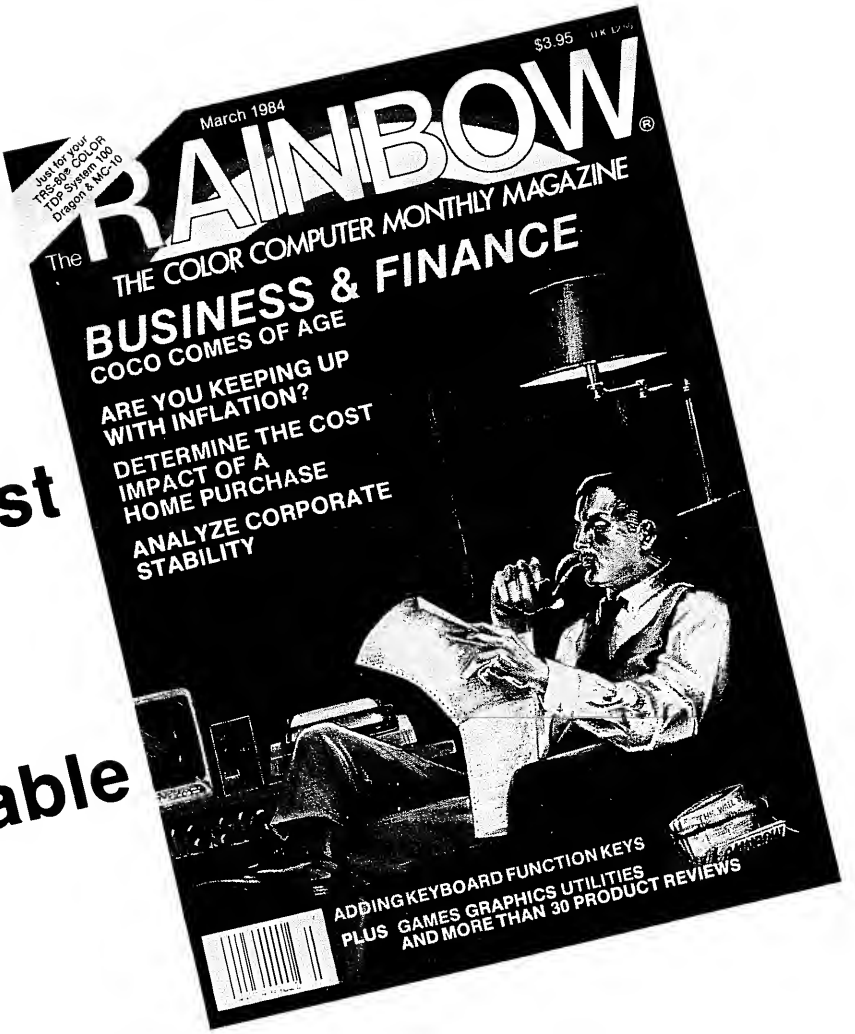
Listing 3. Circle Tester

```
0 REM * CIRCLE TESTER * RICHARD RAMELLA
10 REM * TRS-80 EXTENDED  COLOR BASIC 16K
20 CLS
30 PMODE 4,1
40 PCLS1
50 PRINT "<S>TART-END"
60 PRINT "<H>EIGHT-WIDTH RATIO"
70 PRINT "<R>ADIUS"
80 INPUT C$
90 IF C$="S" THEN 220
100 IF C$="R" THEN 300
110 IF C$="H" THEN 140
120 CLS
130 GOTO 50
140 CLS
150 PRINT "-- 1 TO 255 FOR HEIGHT"
160 PRINT "-- 0 TO .99 FOR WIDTH"
170 INPUT X
180 PCLS 1
190 SCREEN 1,0
200 CIRCLE(128,96),30,2,X
210 IF INKEY$<>"" THEN 140 ELSE 210
220 CLS
230 PRINT "FOR PART OF CIRCLE..."
240 PRINT "-- 2 NUMBERS FROM 0 TO 1
250 INPUT "START,FINISH";X,Y
260 PCLS 1
270 SCREEN 1,0
280 CIRCLE(128,96),40,2,1,X,Y
290 IF INKEY$<>"" THEN 220 ELSE 290
300 CLS
310 PRINT "FOR RADIUS OF A CIRCLE"
320 PRINT "--- 0 TO 65535"
330 INPUT X
340 PCLS 1
350 SCREEN 1,0
360 CIRCLE(128,96),X,2
370 IF INKEY$<>"" THEN 300 ELSE 370
380 END
```

The Biggest

The Best

The Indispensable



The latest news on the Color Computer grapevine is that more and more people are discovering **The Rainbow.**®

Now in its third year, **The Rainbow** has become the standard by which other Color Computer magazines are compared. And no wonder! **The Rainbow** towers above the crowd, now offering up to 300 pages each month, including as many as two dozen type-in-and-run program listings, a host of articles and in excess of 30 hardware and software product reviews.

We lead the pack in Color Computer publications and are devoted *exclusively* to the TRS-80® Color, TDP-100 and Dragon-32. We made our climb to the top by continually offering the best and the most by such well-known authors and innovators as Bob Albrecht and Don Inman, and games from top programmers like Robert Tyson, Fred Scerbo and John Fraysse. **The Rainbow** offers the most in entertainment and education, home uses, technical details and hardware projects, tutorials, utilities, graphics and special features like Rainbow Scoreboard and our CoCo Clubs section.

For only $28* a year, you get the keys to all the secrets locked in your CoCo!

Are you searching through the jungle of claims and clamor? Climb above it all. Look up. Find **The Rainbow.**

THE RAINBOW
(502) 228-4492

9529 U.S. Highway 42
The Falsoft Bldg.
P.O. Box 385
Prospect, KY 40059

THE RAINBOW is a publication of Falsoft, Inc.

YES! Sign me up for a year (12 issues) of **THE RAINBOW.**

Name _____

Address _____

City _____ State _____ ZIP _____

Payment Enclosed ☐

Charge Visa ☐ MasterCard ☐ American Express ☐

My Account# _____

Signature _____ Card Expiration Date _____

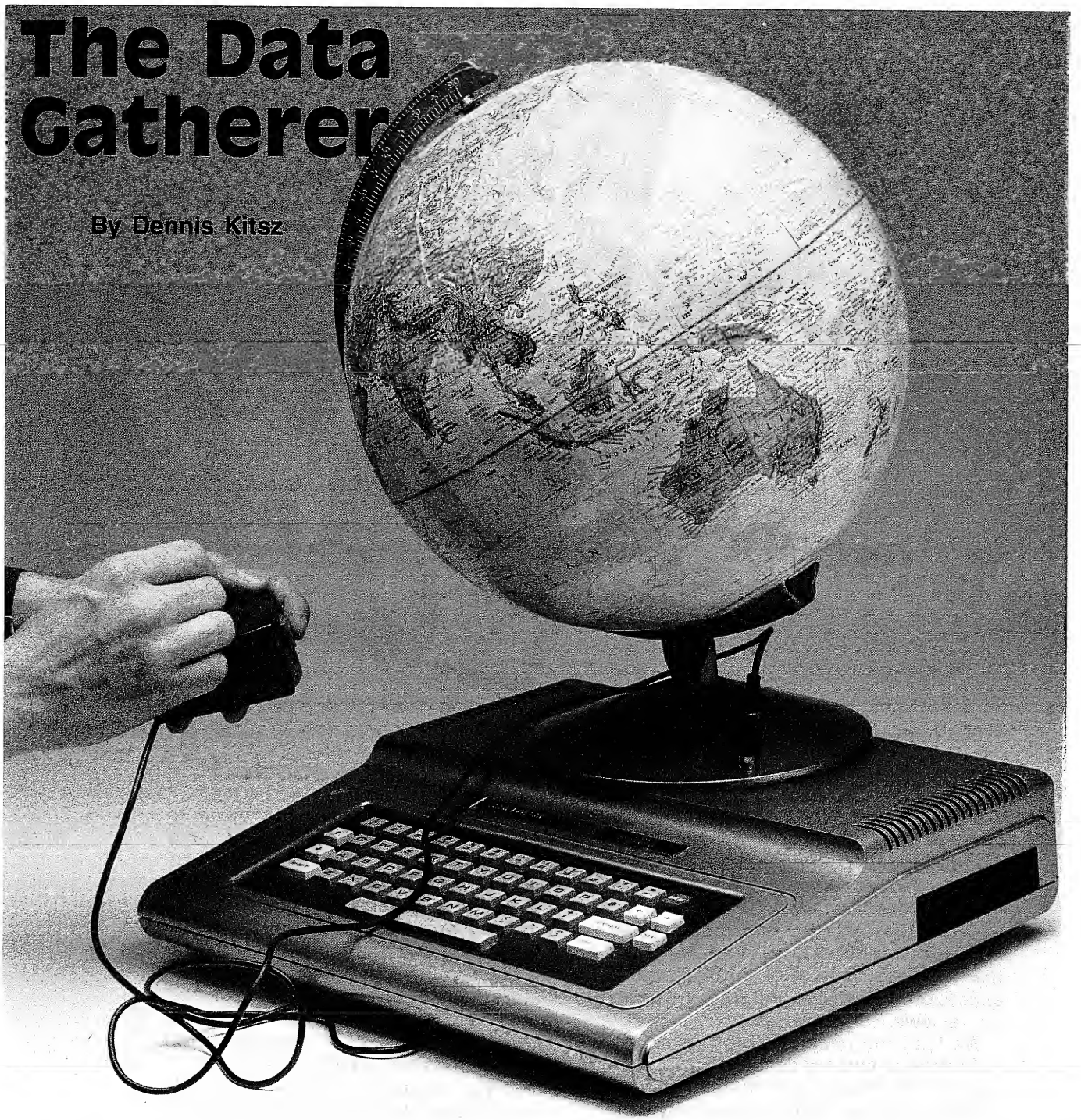
In order to hold down non-editorial costs, we do not bill for subscriptions.

*Subscriptions to **the RAINBOW** are $28 ($31 after Jan. 1, 1985) a year in the United States. Canadian rate $35 U.S. ($38 after Jan. 1, 1985). Surface rate to other countries $65 U.S. ($68 after Jan. 1, 1985); air rate $100 U.S. ($103 after Jan. 1, 1985). All subscriptions begin with the current issue. Please allow up to 5-6 weeks for first copy. U.S. FUNDS only.

Prices subject to change.

The Data Gatherer

By Dennis Kitsz



Things develop in odd ways: in the past two issues I've been working up to a talking real-time clock/calendar, but (how can I say this?) I got distracted. The implications of its battery-back-up real-time clock were powerful for an experimenter like myself, and soon another project drew me away. So for the next few issues you'll be reading about the Data Gatherer, a complete data acquisition system for the Color Computer. (Yes, I will get back to the other talking clock/calendar!)

A data acquisition system is techie jargon for a reasonably accurate, flexible method of interfacing "real world" events to a computer—real signals, real voltages, real time. The Date Gatherer can convert a digital value into a voltage with an accuracy of .025 percent, and can convert a voltage into a digital value with the same accuracy. It can keep the real time, with a battery to keep the time accurate when the computer is turned off, read and set the time, and use the time to trigger computer program segments. It can send or receive digital signals, (in the specific example used here, as a parallel printer port). And, when you turn the power on, its operating system can automatically install and execute a Basic program from read-only-memory (ROM). And finally, it can be used together with other ROM packs or with a disk system.

The Data Gatherer, like the speech synthesizer, makes

use of the interfacing principles and hardware of the CoCoPort (*The Color Computer Magazine*, April–May 1983). The Data Gatherer is also completely modular—you can construct one, two, three or all four of its major sections and use those sections independently.

Analog-to-Digital Conversion

Essential to computer analysis of any non-digital (real world or analog) information is a way to convert that analog information into digital form.

In some cases you've already got digital-like events to analyze, and those demand no special conversion. The number of customers passing through a door, for example, can be counted one at a time (on-off-on-off-on-off). That counting is inherently digital or binary. To record the weight of those customers, however, you would have to measure and convert the weight into computer numbers. A simple electronic alarm system is already digital because you need to detect only whether a window is open (window switch on or off) or if a light beam has been interrupted (beam on or off). On the other hand, in order to determine an intruder's exact position in a building, you must measure and convert a value into computer terms.

What if you wanted to keep track of the electricity coming into your home? Knowing if the electricity is on or off is a digital event; but measuring the actual voltage is more difficult, and that's a similar problem to how joysticks are used.

Atari-style joysticks are four simple on-off switches. Each switch determines one direction or none—up, down, left, right, off—and any two neighbors can both be clicked on—up-left, up-right, down-left, down-right. Radio Shack joysticks are more sophisticated, permitting not just the nine positions Atari allows, but 64 unique positions in each direction, a total of 4,096 distinct locations. (Refer to Issue 1 for more details about the Color Computer joysticks.)

To measure an actual voltage, then, you've got to come up with an analog-to-digital conversion process that's accurate enough for whatever you need to measure.

There are two major ways of performing the conversion—hardware-intensive or software-intensive. (Sound familiar? Computer applications *always* involve trade-offs between hardware and software solutions.) The hardware-intensive solution uses a complete analog-to-digital integrated circuit which, when kicked by the computer, measures voltage, and a few microseconds later reports the result to the computer. This is fast, effective—and costly. Complete 12-bit (.025 percent accurate, or one part in 4096) analog-to-digital converters cost $150 or more each.

With the exception of its greater attention to detail, the software-intensive solution for accurately measuring an analog value is identical to the joystick input process. The computer software calculates and outputs a value to a digital-to-analog converter (much cheaper—about $20). The value is altered and compared via software until the computer finds a match. The Data Gatherer uses this solution, called successive approximation.

Digital-to-Analog Conversion

At the heart of the Data Gatherer is a digital-to-analog converter. The governing principle of such converters is binary "weighting"; that is, each binary digit of a binary word has a value, or weight, twice that of its neighbor. Here's a simple example:

In decimal terms, the binary word 000001 is 1, 000010 is 2, 000100 is 4, 0010000 is 8, 010000 is 16 and 100000 is 32. Each position is twice that of the previous. Complex binary words can be converted to decimal form by summing each of the individual bit values: 101110 is 32+0+8+4+2+0, or 46.

This same mathematical conversion can be applied to converting binary numbers into voltage. Hook the computer to a "black box." The black box assigns values to each bit, based on its "weight" within the binary word. That is, if the right-most binary digit is worth one, then the black box outputs 1/64th of a volt; the next binary digit is worth two, and the black box assigns it 2/64ths of a volt; all the way up the left-most binary digit, worth 32, and assigned 32/64ths of a volt. The binary example above (101110) then becomes the sume of the individual binary digits (46), and the black box outputs 46/64ths of a volt.

The original grey-case Color Computer had the simplest kind of digital-to-analog (D/A) converter, known as a resistor ladder. See how, in the sketch below, each resistor is twice the value of the next, meaning the voltage getting through each resistor is only half the value of its neighbor. The far ends of all six resistors are wired together, resulting in a sum of all the weighted voltages. In the Color Computer, not one volt, but rather five volts are sent to the top of the resistor ladder, meaning the output voltage is divided (scaled) into 5/64ths of a volt (.078 volts) per binary digit of measurement.

In summary, then, a six-bit digital-to-analog converter has a scale running from 0 volts (binary 000000) to 4.92 volts (binary 111111) in steps of .078 volts (1 bit). Take a look at Figure 1.

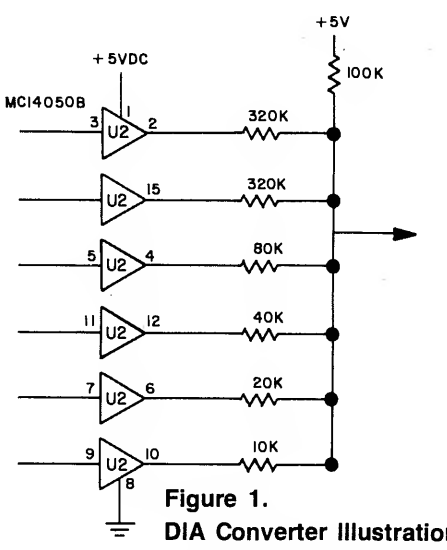


Figure 1.
DIA Converter Illustration

To get good precision (even at six bits), the resistors must be accurate to within one bit or less (better than one part in 64). This means that for reasonable precision the resistors used in the ladder should be 1 percent types (one part in 100 accuracy). A check of the Color Service Manual reveals that, yes indeed, Radio Shack used 1 percent resistors—10K, 20K, 40.2K, 80.6K, 162K and 324K.

The Color Computer has only a 6-bit D/A converter, but the Data Gatherer uses a 12-bit D/A converter. What does this mean? It means that the least significant bit of the Data Gatherer's converter is 1/4096th of a division—an accuracy

of .0244 percent. On the same scale of the Radio Shack machine, this means the scale runs from 0 volts (0000 0000 0000) to 4.9987 volts (1111 1111 1111), with each division worth just .0012 volts.

Inside the Data Gather's converter (an Analog Devices type AD667) is a simple resistor ladder. But since Analog Devices states that the AD667 is accurate to within 1/2 bit, it means they have produced resistors with an accuracy of at least .012 percent—100 times more accurate than the 1 percent precision resistors in the Radio Shack machine. This is done by "laser trimming" each resistor until it meets the manufacturer's specifications, a process that drives the cost up quickly.

Just for comparison, the total cost of the parts in the Radio Shack computer's D/A converter (seven resistors, one CMOS butter) is about 20 cents in quantities of 100. The Analog Devices AD667s cost $20 or more each in quantities of 100—100 times the accuracy, 100 times the cost.

A block diagram of the AD667 is presented in Figure 2. I'll discuss later the details of the 4-bit latches shown; beyond that, notice the complete simplicity of the device . . . a latch to hold the digital data accurately in place during conversion, level translator and resistor ladder (marked 12-bit high speed DAC), and a scaling amplifier. The amplifier allows the user to select the full voltage scale across which the D/A converter can be used: 0 to 5 volts or 0 to 10 volts. Later I'll give you a few application ideas for using the two scales.

Real-Time Clock/Calendar

I am planning to save this one for the talking clock/calendar project, and so you'll get very little of the theory and ideas behind the clock/calendar itself right now. Instead, I'll present an interesting hardware trick for fitting a lot of trinkets into a little box. (If you're interested and absolutely must know about the clock/calendar, take a look at Figure 3 for an idea of how thoughtfully designed this $15 microprocessor-compatible clock/calendar is.)

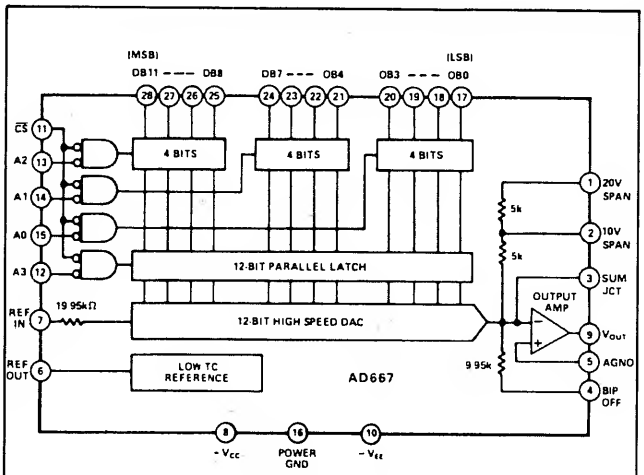


Figure 2. Block Diagram—AD667

To implement the clock in a full-blown Color Computer system you need some background about its unique arrangement. The Color Computer uses a 6809E microprocessor, its brain, which can address 64K (65,536 bytes) of memory. When this microprocessor is paired with a special memory management circuit known as the Synchronous Address Multiplexer (SAM), that map of 64K bytes is organized and expanded. That expansion includes the ability to address 96K (98,304 bytes) of memory—and possibly more than that in newer versions of the SAM.

Of particular interest is how the SAM organizes the top page (highest 256 bytes) of memory. This small section of memory has been cordoned off for control purposes. Briefly, that involves signals for choosing graphics modes, using the keyboard peripheral interface adapter (PIA), using the cassette

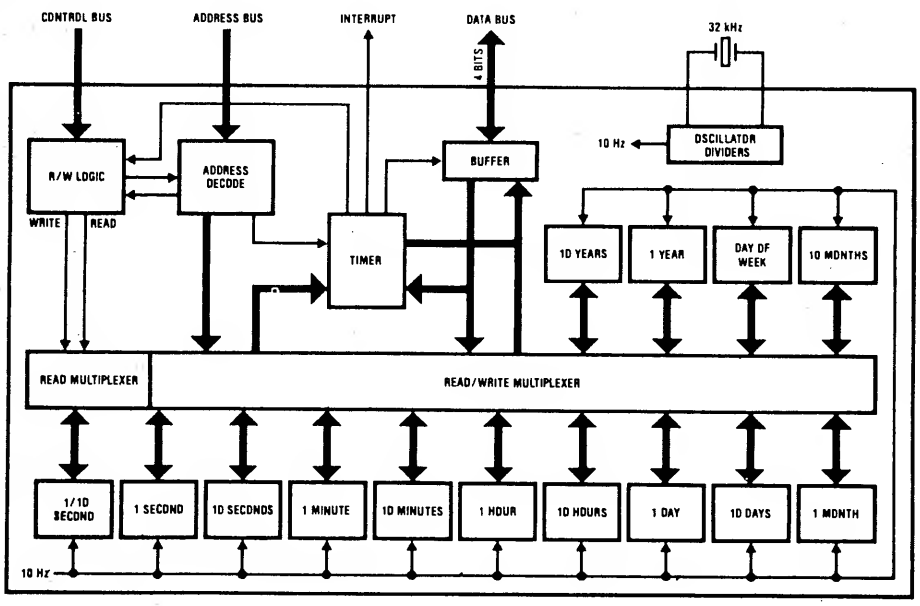


Figure 3. Block Diagram

motor and joystick PIA, etc. One signal from the SAM (labeled SCS*, or not-spare-chip-select) is reserved to distinguish a block of 32 addresses in memory from location $FF40 to $FF5F (decimal 65344 to 65375).

It is this block of 32 bytes that the Radio Shack disk pack uses to read and write, select, and run the motor of your disk drives.

It also turns out that a standard disk drive doesn't actually need all 32 of these addresses, although (for unknown reasons) Radio Shack chose to identify the 16 addresses from $FF40 through $FF4F are reserved in its software. Nevertheless, by cleverly demultiplexing (decoding) the block $FF40 to $FF5F, the large 32-byte block can be split into one 16-byte section for the disk and one 16-byte section for your own use. These latter 16 bytes can be further multiplexed into four groups of four. Such is the principle of the CoCoPort.

The Data Gatherer is going to need some of those spare 16 addresses for the digital-to-analog converter for the analog-to-digital converter and its channel-selection process (more on that next time), as well as for a printer port.

But a problem arises. According to the manufacturer's data, the real-time clock/calendar alone requires all 16 addresses; in Table 2, you can see that 14 addresses (1–14) are used for reading from tenths of seconds of tens of years, and the two remaining addresses (0 and 16) are used for controlling the clock/calendar's operation and for setting the clock.

How can the D/A converter, A/D converter and printer have *their* 16 addresses if the clock/calendar wants to hog them all? Or conversely, if we don't put the clock there, how can the disk drive have *its* 16 addresses? Examine the Data Gatherer block diagram (Figure 1). The "decode" area is the $FF40–$FF5F demultiplexer, and it selects respectively: a PIA for the D/A converter; a PIA for the A/D converter input and printer; the real-time clock; and (immediately below the block marked "R/W* logic") the disk drive selector.

Look at the "R/W* logic" part of the diagram. This section directs a block of 16 addresses either to the disk drive or to the clock/calendar. It isn't done with mirrors, but it's close.

Fortunately, all the rest of the Data Gatherer's contraptions won't require 16 addresses (actually only eight) so there are a few free. One of these free addresses is a kind of "which-one" selector. By storing a zero at $FF58 (decimal 65368), you permit the disk to be used; by storing a one at $FF58, you permit the clock to be used. The circuit is a simple flip-flop, triggered by the data written to $FF58: flip, it's the disk, flop, it's the clock.

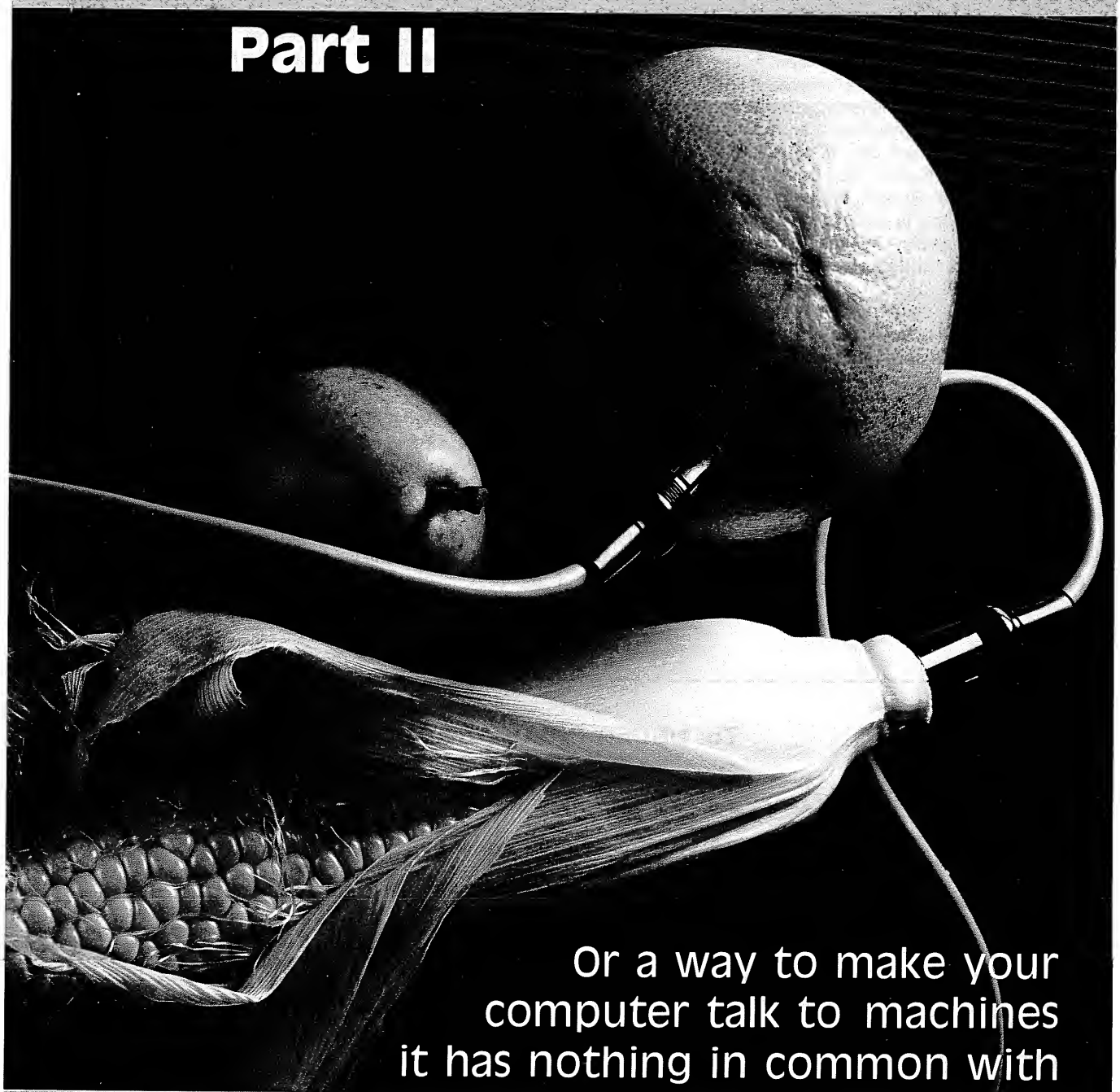
Generalize this process. It is the heart of "bank selection," a term which you will hear applied to many things, from choosing among various peripherals (Radio Shack's Multipak is a kind of heavy-handed hardware bank selector) to turning on one among several large blocks of memory (RGS Micro's 128K adaptor for the CoCo requires bank selection to give you that much RAM).

Your "32K/64K" Color Computer is a bank-selected machine, too. When you are using Basic, you have 32K RAM, 8K ROM (Color Basic), 8K ROM (Extended Basic), 8K ROM (Disk Basic), and 8K unused . . . a total of 64K. When you are using a special 64K RAM program such as a word processor (VIP Writer, TeleWriter, etc.), the three Basic ROMs are turned off and the upper 32K RAM is turned on in their place. That's bank selection.

So to fit all these electronic contraptions into one little block of available memory, you have to resort to the type of fancy footwork called bank selection. As I said, there will be more about the clock itself when I pick up the talking clock/calendar project in a few weeks. *(end)*

CROSS TALK

Part II



Or a way to make your computer talk to machines it has nothing in common with

By William Barden, Jr.

Last month, Bill explained what RS-232C is, told us about bauds, and presented some programs to use. This month he tells us what we can do with them! Refer to part one for the programs.

You can easily use the program to connect two Color Computers and pass data between them. Prepare a cable as follows:

Get two 4-pin DIN connectors from Radio Shack (274-007). Solder a three-wire cable between the two connectors, as shown in Figure 1. In this configuration the TD signal from one side goes into the RD signal for the other system, and vice versa.

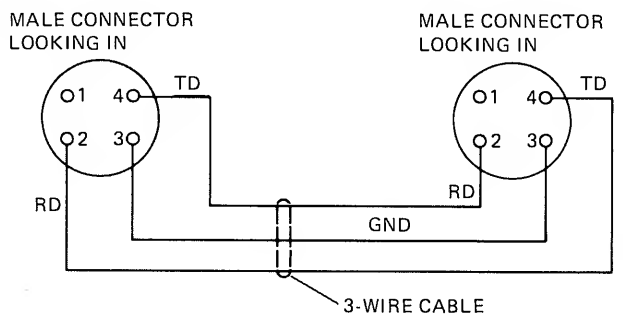


Figure 1. Color to Color Connections

Load the program into both computers. Start the receiving system by calling the Input Character routine after POKEing the starting and ending locations of the memory area to be filled with data (or use the screen addresses). Next, start the transmitting system by calling the Output Character routine from Basic. If you're using the terminal program above, you can start either system first.

If you're using the program to transfer data between your Color Computer and another computer system, you may have some difficulties—not on the Color Computer end, but in the connection with the other system. Other computer systems use additional signals beyond the 22 RS-232C signals to establish a protocol. There are no clear-cut rules by which signals are used. It may be necessary to "dummy up" some of the signals on the other system to fool it into thinking the signals are being properly received from the Color Computer. This is done by routing an output signal to an input signal on the same end. It may be necessary to do this for several signals, and you may have to wire a special DB-25 plug to accomplish this. A convenient plug to use is the Radio Shack male DB-25 (276-1559). Regardless of the signals dummied up on the other end, there need be only three signals on the Color Computer end. A typical configuration is shown in Figure 2.

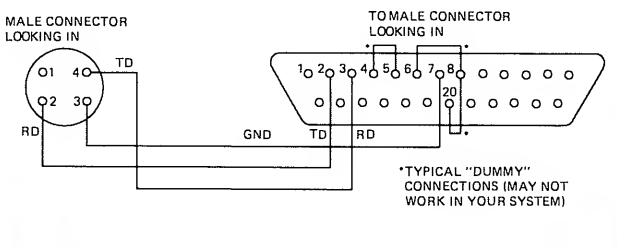


Figure 2. Color to Other Computer Connections

To Peripheral Device

You can also use this program to communicate with peripheral devices, such as printers and plotters. For reference only, the line printer-driver subroutine in ROM uses three lines, but not in the same way we've been connecting devices here. The RD line is used as a status line to indicate whether the line printer or other device is ready to receive data, as shown in Figure 3. This line must normally be kept at a voltage level of greater than +3 volts, or must be connected to an appropriate status signal from the device.

If you're using the Character Output routine to transmit data to a printer you'll need only the TD and ground lines. Connect the TD line to the RD printer input and the ground line to the ground printer line. Again, it may be necessary to dummy up some of the other lines on the printer end, as shown in Figure 4.

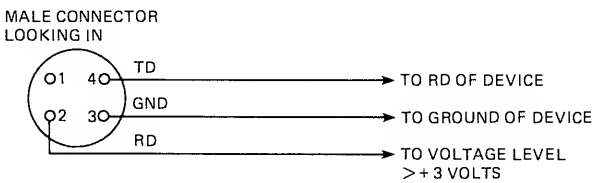


Figure 3. Line Printer Connections

Data-Buffer and Overrun

Using the RD line as a status indicator is an excellent idea. Don't forget—if you're sending data to a peripheral device, that device must be able to keep up with the data transmission rate. If a printer operates at 180 characters per second, for example, and you're sending data at 2400 baud (240 characters per second for 10-bit characters), you'll quickly overrun the printer. Most printers or other peripherals have some buffering that allows the device to save a limited number of characters, usually hundreds. Buffering lets the device receive bursts of data at high-speed rates, and then empty the buffer (a temporary storage spot) during periods when no transmissions are taking place.

If data overruns the buffer, data following the overrun will be lost. To overcome this problem, other RS-232C lines, such as RTS (request to send), are typically used, but our program makes no provision for a return of status.

You might notice in the Basic terminal program that characters are occasionally lost. This usually occurs for repeated characters, such as the second O in good; it happens because of the speed restrictions of the Basic keyboard scan routine. Characters sent from within a program will not be lost.

We've barely scratched the surface of RS-232C. Use the program to experiment with your own data communications applications. At another time I'll show you how a slight modification of the basic program can be used to implement your own terminal program for communication with CompuServe and other data communications networks. *(end)*

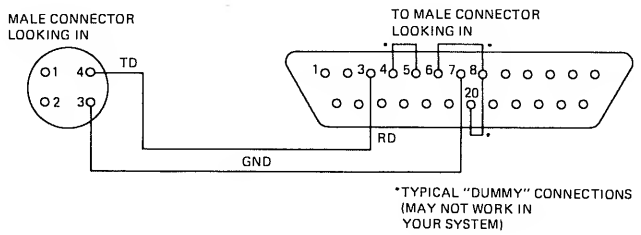


Figure 4. Printer Connections For RS-232C Program

The assembly language program really just implements what we've discussed here. On the output side, a zero bit is output to PIA $FF20 for one bit time. Next, the number of specified data bits are output through the $FF20 PIA a bit at a time, starting with the least significant bit. Finally, a one stop bit is output to PIA $FF20 for one or two bit times. The output on the TD line is left in the stopped state.

On the input side, the RD line is monitored by reading PIA $FF22. Whenever a zero bit is detected, the program starts counting by a delay loop based on the baud. After 1 ½ bit times the first data bit is read. The remaining bits are read and assembled into 7- or 8-bit characters. After the last data bit has been read, the program delays for the following stop bits.

Two delay loops delay either one bit time (DELAY) or one-half bit time (DELAYH). A delay count is obtained from a baud table indexed by the baud code (0–7). Note that the delay counts are slightly different for output and input, reflecting the slightly different numbers of instructions for each operation. As each instruction executes in a fixed time, the delay count is determined by considering the time through the program delay loop plus some additional "overhead" for other operations. You may try changing the table values slightly to alter the bit times if you add additional assembly language code in the program. Send constant characters to determine minimum and maximum values for error-free transmissions, and then choose an intermediate value for each baud.

The keyboard scan in the Input Character routine quickly reads all keyboard columns to determine if a key has been pressed. Using the Basic ROM subroutine proved to be too slow to avoid data overrun at higher bauds. Similarly, using the ROM subroutine for Display Character proved to be too slow at the end of a line when the screen was scrolled. Several characters are lost during the scrolling time.

Hints, Tips, and Tricks

By Terry Kepner

Printer—To route all normal text output from the screen to a printer, type POKE 360,162:POKE 361,191. If you are in Color Basic, add POKE 359,126. To restore normal print operation, type POKE 359,57 in Color Basic: POKE 360,130:POKE 361,115 in Extended Color Basic; or POKE 360,203:POKE 361,74 in Disk Basic.

Programming—To protect video information from scrolling off the screen, POKE 359,16 for a cassette system, or POKE 359,0 for a disk system. After this POKE only PRINT @s and POKEs to the video memory will show up on the screen. To restore normal operation, type POKE 359,126.

Printer—To check your printer's on/off status, PEEK (65314) and PEEK (65318). If you get four, everything is OK, but a five or seven indicates the printer is off, or offline.

Disk Basic—To turn your drive motors on, type POKE &HFF40,60. To turn them off, type POKE &HFF40,0. This eliminates the wait time before reading and writing data to your disks. The drives are also turned off by normal disk commands.

Programming—For less consistent random numbers (RND function always starts with the same sequence), add this to the start of your programs: POKE 280,PEEK(275). Location 280 is one of the RND seed addresses, and location 275 contains a constantly changing number.

Disk Basic—To disable Disk Basic without unplugging the ROMpak cartridge, type POKE 298,0:POKE 303,0. This makes Basic ignore all the Disk Basic commands. To restore normal operation, type POKE 298,25:POKE 303,14. Reset won't do the job.

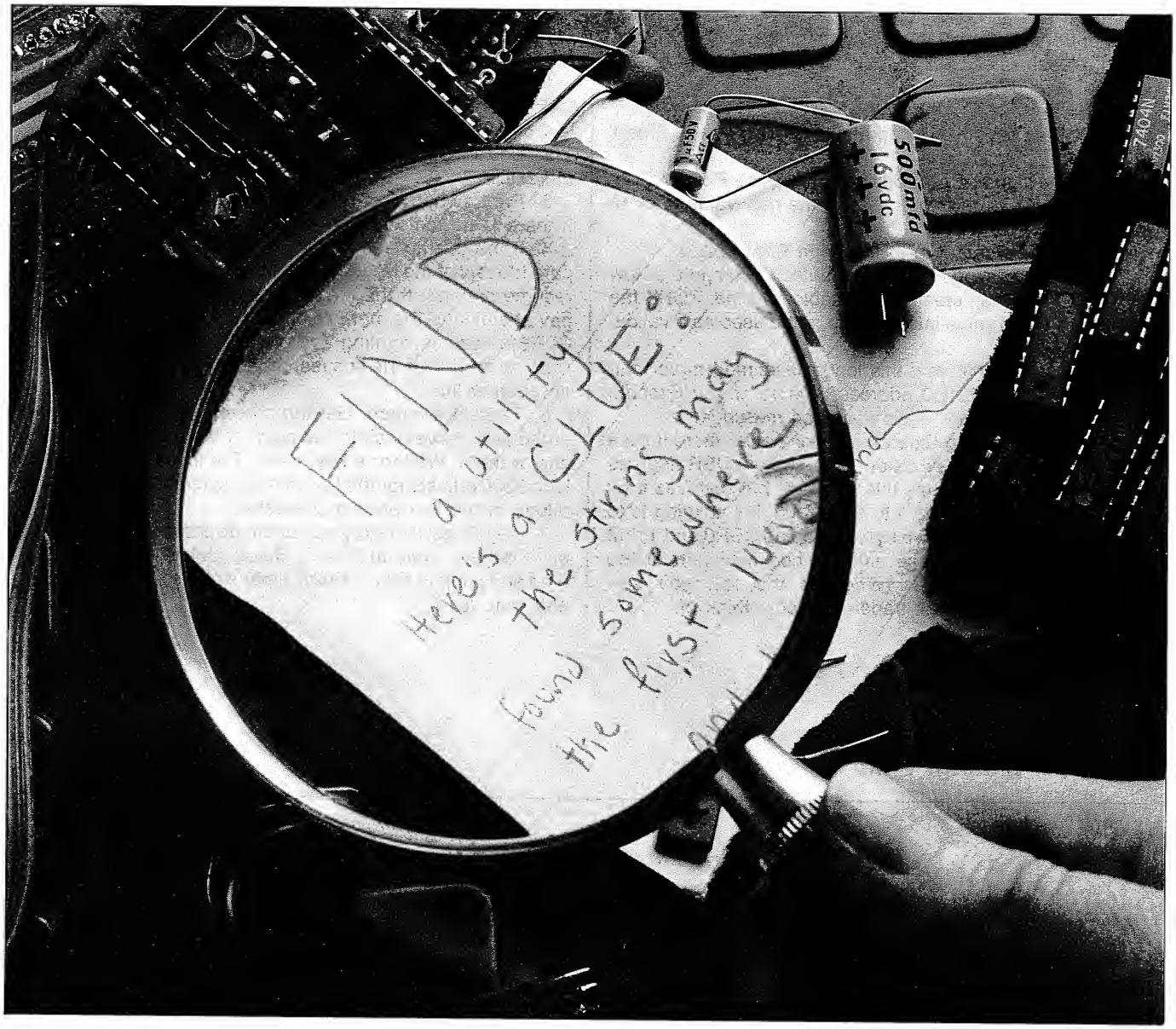
Disk Basic—To get a printed copy of your disk directory and the remaining free granules, type POKE 111,254:DIR:PRINT#-2, "Free="];:PRINT#-2, FREE(0). This can, of course, be put into a program.

General—To clear all memory and cold start your computer, type POKE 113,3:EXEC 40999.

ROMpak—To dump a ROMpak to cassette, put a piece of tape over pin 1 of the ROMpak (the leftmost pin as you look at the ROMpak connector, label up), plug it into your computer, and turn it on. Set up your cassette recorder and type CSAVEM"filename",&HC000,&HFEFF,&HC000.

Printer—The pens on the Radio Shack CGP-115 printer plotter dry out if not used every day or two. If you turn it on every day, the start-up sequence is enough to keep the ink flowing evenly. The best idea is to buy a power strip and plug all your equipment into it. Whenever you use your computer, your printer will automatically go through its start-up sequence without your having to do it explicitly.

Disk Basic—To protect your disk programs/files from unauthorized access, save them with a graphics code embedded, i.e., SAVE"file"+CHR$(143). The file will list in the directory, but you can't reload it unless you use the same method as you used to save it: LOAD"file"+CHR$(143). CHR$(143) is a graphics space, so no one will know it's there but you!



By Stephen P. Allen

Remember "System Horizons" (*TCCM,* July '84)? "Find" is a follow-up to that article, and it has finally found a home!

To disassemble Basic ROMs; to be an "insider" on the real processes taking place in your computer; these impart fun and satisfaction, but more satisfying is taking Basic's own subroutines and using them to create your own programs. You'll get a great deal of power from a very small piece of code.

What does Find do?

Say you've been working on a long Basic program and you keep getting ?FC ERROR in a line you know is legal. Suddenly you realize you need to change a single occurrence of N$(A) to N$(B). You can do nothing but List the program, hitting Shift @ 'til you find it (*if* you fint it; oops! Try again . . .), then Break and Edit the correct line. A drag, right?

The Find utility lets you type FINDN$(A) and enter. A listing instantly appears and scrolls furiously by, just as in List. When a line containing the target string is listed, however, the display freezes and a gentle blue cursor marks (in this case) N$(A)'s position in the line.

At this point you have three options. You can hit Break and break out of Find, or you can hit Enter and continue the listing and search. Or (and this is the best part), you can hit E and go immediately to Extended Color Basic's Edit routine.

Want to search for the same string a second time? Just type FIND and enter. Find has its own 10-character buffer, so that's your maximum target string length.

How about wild card searches? Sure! The symbol is #. Searching for PR##T will locate PRINT, PRESTO, PR$(T), etc.

If you're searching for a string containing a Basic word, use a single quotation mark before the word. For example: FIND"PUT. The quotation mark keeps Basic from turning the string into a token, and is good practice generally, since (for example) NONE without the mark is tokenized into N/ token for ON/E. Find understands the quotation mark and skips it when it's the first character.

If your Basic program is already in memory and you need to load Find, don't worry. Find has two parts: the guts, which contain the routine, and the Pretty Smart Loader, used only once, which sets up operation of the routine. Together, they load into low graphics memory. Even if you've executed PCLEAR1 you're OK. Just (C)LOADM and Execute to set it up, and you're in business.

All this power takes only 250 bytes!

Details

Note: from here on, if there's something you don't understand, please refer back to "Exploring System Horizons" (TCCM, *July '84*).

Here's the sequence of operations for **The Pretty Smart Loader:**

1. Get the highest RAM address from Basic's Pointer. Subtract the length of the "guts." Give the new value back to Basic.
2. Copy the "guts" up to that high RAM space.
3. Use the existing RAM token tables to compute a new token for Find, then create a new table for Find. POKE the calculated token value into the guts, to be used as a validation check.
4. Since we don't want this loader executed more than once, change the EXEC address to 41175. If you Execute again, you'll get Color Basic's sign-on message.
5. If you're running Extended Color Basic, there'll be a conflict between the new token table and the USR address table. Disk Basic also has this problem, but resolves it by moving the USR table up into Disk RAM. My solution is to move the USR table up two bytes, keeping USR0 and 1, but taking away USR9, whose address becomes part of the keyboard roll-over table! (When is the last you saw DEFUSR9?) It's the best trade-off I could think of.

The Guts: Find is always entered with its token in the A register. A quick check is done to see if Basic got there by mistake. If everything's OK (always is), then proceed.

1. Evaluate and store the target string. This is the string which Find will try to match. If no string is specified, use the last string.
2. Use a ROM call to de-tokenize the next line into a buffer at $2DD.
3. Print a character from the buffer, testing for a match. If there is no match, keep printing and checking.
4. If the character matches, suspend printing and compare the rest of the target string. If the rest of the match fails, resume printing. If the match is successful, set a flag and pay attention to the match's position in the line (Line 2100, CLRB). Resume printing and checking.
5. At the end of the line test the flag. If there is no match, do another line.
6. There's a match. Get the screen position of the end of the line, move back to the point of the match, and set a cursor there. Wait for a key press. The instruction INCB in Line 2300 adjusts for the fact that the cursor points one byte ahead of the last printed character.
7. Key Press: If the key was Enter, do another line. If Break, exit Find and jump to Basic's Break routine. If E, then exit Find and jump to Edit in ROM. If none of the above, go back and wait. *(end)*

Program Listing. Find

```
                  00100 ************************
                  00110 **                    **
                  00120 **        FIND        **
                  00130 ************************
                  00140 **    A New Word For  **
                  00150 **        Basic       **
                  00160 ************************
                  00170 *          by          *
                  00180 *    Stephen P. Allen  *
                  00190
                  00200
                  00210
                  00220
                  00230 * For Disk Basic use this ORG
0F00              00240         ORG     $F00
                  00250
                  00260 * For Extended Basic use this ORG
                  00270 *       ORG     $700
                  00280
                  00290 ************************
                  00300 *         THE          *
                  00310 *     PRETTY SMART     *
                  00320 *        LOADER        *
                  00330 ************************
                  00340
                  00350
                  00360
                  00370 *** MAKE ROOM UP TOP ***
                  00380
         00F9     00390 LENGTH  EQU     ZEND-TITLE
                  00400
0F00 DC   74      00410 START   LDD     <$74    High Ram address
0F02 83   00F9    00420         SUBD    #LENGTH -Find's length
0F05 DD   74      00430         STD     <$74    =New "High Ram" address
0F07 DD   27      00440         STD     <$27     and "Top of String Space"
0F09 DD   23      00450         STD     <$23     and pointer to next string
                  00460
0F0B 83   00C8    00470         SUBD    #200    default total string space
0F0E DD   21      00480         STD     <$21    to "Start of String Space"
0F10 10DE 21      00490         LDS     <$21    Stack goes below strings
                  00500
                  00510 ***    MOVE IT UP    ***
                  00520
0F13 DE   74      00530         LDU     <$74    New "Top of Ram"
0F15 33   42      00540         LEAU    2,U     (a little space, just in case)
0F17 30   8C 41   00550         LEAX    <TITLE,PCR    Start of "FIND"
0F1A 108E 00F9    00560         LDY     #LENGTH         of "FIND"
0F1E A6   80      00570 PUTHI   LDA     ,X+
0F20 A7   C0      00580         STA     ,U+
0F22 31   3F      00590         LEAY    -1,Y
0F24 26   F8      00600         BNE     PUTHI
                  00610
                  00620
                  00630 *** TELL BASIC WHERE TO FIND IT ***
                  00640
0F26 CE   0116    00650         LDU     #$116   Address of (token tables-10)
```

```
0F29 5F              00660          CLRB             Token will form here
                     00670
                     00680
0F2A 33   4A         00690 TABLE    LEAU    10,U     Move into next Token Table
0F2C EB   C4         00700          ADDB    ,U       add # of entries this table
0F2E 6D   C4         00710          TST     ,U       At end of tables yet?
0F30 26   F8         00720          BNE     TABLE    no, keep going
                     00730
0F32 CB   80         00740          ADDB    #$80     Make negative, signal as token
0F34 6F   4A         00750          CLR     10,U     Signal new "End of Tables"
0F36 86   01         00760          LDA     #1       One new entry here
0F38 A7   C0         00770          STA     ,U+      At start this new table
                     00780
                     00790 *  Recalculate start of "FIND"
                     00800
0F3A 9E   74         00810          LDX     <$74     High Ram address
0F3C 30   02         00820          LEAX    2,X      plus that two (just in case)
0F3E AF   C1         00830          STX     ,U++     Address of word into Table
                     00840
0F40 30   04         00850          LEAX    4,X      Address of routine...
0F42 AF   C4         00860          STX     ,U       into the table
                     00870
                     00880 * New token also goes directly into routine *
                     00890
0F44 E7   01         00900          STB     1,X
                     00910
                     00920 *Change EXEC address to disable Loader
                     00930
0F46 8E   A0D7       00940          LDX     #41175   Color Basic's Sign-on msg.
0F49 9F   9D         00950          STX     <$9D     to EXEC pointer
                     00960
                     00970
                     00980 *Move DEFUSR table if necessary*
                     00990
0F4B DC   B0         01000          LDD     <$B0     Points to DEFUSR table
0F4D 1083 013E       01010          CMPD    #$13E    Extended Basic's table?
0F51 26   04         01020          BNE     JUMP     if not, then Disk: OK
0F53 C6   40         01030          LDB     #$40     Else Make new table at $140
0F55 DD   B0         01040          STD     <$B0
                     01050
0F57 6E   9F FFFE    01060 JUMP     JMP     [$FFFE]  Done! Go do a warm start.
                     01070
                     01080
                     01090 ************************
                     01100 *        THE GUTS      *
                     01110 ************************
                     01120
                     01130 *  "FIND" with "D" made negative
                     01140
0F5B      46         01150 TITLE    FCC     /FIN/
          49
          4E
0F5E      C4         01160          FCB     $80+'D
                     01170
                     01180
                     01190 *   Routine starts here  *
                     01200
0F5F 81   CE         01210 FIND     CMPA    #$CE     If disk, this will change
0F61 27   03         01220          BEQ     OK       yes, correct token
0F63 7E   B277       01230          JMP     $B277    else ?SN ERROR
                     01240
                     01250
                     01260 *** GET THE PARAMETERS ***
                     01270
0F66 9D   9F         01280 OK       JSR     <$9F     next character
0F68 27   1C         01290          BEQ     FIND2    if null, use previous string
                     01300
                     01310 * A string is specified
                     01320
0F6A 30   8D 00DC    01330          LEAX    TARGET,PCR   Point to buffer
0F6E 5F              01340          CLRB             Length of string so far, 0
0F6F 81   22         01350          CMPA    #'"      String delimiter?
0F71 26   04         01360          BNE     LOOP1
0F73 9D   9F         01370          JSR     <$9F     if so, skip it
0F75 27   0F         01380          BEQ     FIND2    if nothing else specified
                     01390
0F77 A7   80         01400 LOOP1    STA     ,X+      character into FIND's buffer
0F79 5C              01410          INCB             length=length+1
0F7A C1   0A         01420          CMPB    #10      max # of chars?
0F7C 24   04         01430          BHS     DONE     if yes, skip the rest
0F7E 9D   9F         01440          JSR     <$9F     else get next char.
0F80 26   F5         01450          BNE     LOOP1    if another char, loop back
                     01460
0F82 E7   8D 00C3    01470 DONE     STB     LENTS,PCR     Store string length
                     01480
0F86 32   62         01490 FIND2    LEAS    2,S      Drop return address
0F88 8E   0019       01500          LDX     #$19     Has address of first line
0F8B 9F   66         01510          STX     <$66     Save for LIST loop
                     01520
                     01530
                     01540 *** LIST A LINE: TEST FOR MATCH ***
                     01550
0F8D BD   B95C       01560 NXTLIN   JSR     $B95C    Do a linefeed
0F90 BD   A549       01570          JSR     $A549    Test for <Break>, <shift @>
                     01580
0F93 9E   66         01590          LDX     <$66     get address of last line
0F95 AE   84         01600          LDX     ,X       Point to next line
0F97 EC   84         01610          LDD     ,X       End of Basic program?
0F99 26   03         01620          BNE     CONT     if not end of Basic program
0F9B 7E   AC73       01630          JMP     $AC73    Exit here if End of Program
                     01640
0F9E 9F   66         01650 CONT     STX     <$66     Update Low Ram Line Pointer
0FA0 6F   8D 00A2    01660          CLR     FFLAG,PCR     No matches this line yet
                     01670
0FA4 EC   02         01680          LDD     2,X      Get line number
0FA6 BD   BDCC       01690          JSR     $BDCC    Print line number
0FA9 BD   B9AC       01700          JSR     $B9AC    print a space
                     01710
0FAC 9E   66         01720          LDX     <$66     Point to Basic line again
0FAE BD   B7C2       01730          JSR     $B7C2    De-Tokenize into buffer
0FB1 CE   02DD       01740          LDU     #$2DD    Point to buffer
0FB4 31   8D 0092    01750          LEAY    TARGET,PCR   Point to Target string
```

```
                        01770
0FB8 A6   C0            01780 PRINT   LDA     ,U+       Next char. from buffer
0FBA 27   2E            01790         BEQ     TEST      if End of Line
0FBC AD   9F A002       01800         JSR     [$A002] Print the next character
0FC0 5C                 01810         INCB              Keep track of match position
0FC1 A1   A4            01820         CMPA    ,Y        Match on first letter?
0FC3 26   F3            01830         BNE     PRINT     if not, keep going
                        01840
                        01850 * MATCH ON FIRST LETTER *
                        01860
0FC5 34   64            01870 MATCH1  PSHS    U,Y,B     Save important registers
0FC7 33   5F            01880         LEAU    -1,U      Prepare for following loop
0FC9 E6   8D 007C       01890         LDB     LENTS,PCR   Length of Target string
0FCD 5A                 01900 MATCH2  DECB              One less Target char to match
0FCE 27   12            01910         BEQ     FOUND     If match on total string
0FD0 33   41            01920         LEAU    1,U       Else move up in buffer...
0FD2 31   21            01930         LEAY    1,Y        and Target buffer
0FD4 A6   A4            01940         LDA     ,Y        Next target character
0FD6 81   23            01950         CMPA    #'#       Is it a "Wildcard?"
0FD8 27   F3            01960         BEQ     MATCH2    if yes, automatic match
0FDA A1   C4            01970         CMPA    ,U        else test for match
0FDC 27   EF            01980         BEQ     MATCH2    so far so good
                        01990
                        02000 *Match fails in mid-string
                        02010
0FDE 35   64            02020         PULS    B,Y,U     Restore crunched registers
0FE0 20   D6            02030         BRA     PRINT     and keep going
                        02040
                        02050
                        02060 *** FOUND A MATCH ***
                        02070
0FE2 6C   8C 61         02080 FOUND   INC     <FFLAG,PCR   Flag a match on this line
0FE5 35   64            02090         PULS    B,Y,U     restore crunched registers
0FE7 5F                 02100         CLRB              Print distance from this
                        02110 *                            match=0
0FE8 20   CE            02120         BRA     PRINT     Keep listing to End of Line
                        02130
                        02140
                        02150 *   END OF LINE:
                        02160 *   IF THERE'S A MATCH THEN SHOW IT,
                        02170 *   ELSE DO NEXT LINE
                        02180
0FEA EF   8C 5A         02190 TEST    STU     <BUFEND,PCR   Save in case of EDIT
0FED 6D   8C 56         02200         TST     <FFLAG,PCR   Any matches?
0FF0 27   9B            02210         BEQ     NXTLIN    if not, do next line
                        02220
                        02230
                        02240 *   GOT A MATCH:
                        02250 *   PUT CURSOR AND WAIT FOR KEYPRESS
                        02260
                        02270 *   First, move cursor to start of last match
                        02280
0FF2 DE   88            02290 SHOWIT  LDU     <$88      Get current cursor position
0FF4 5C                 02300         INCB              Adjust for offset
0FF5 33   5F            02310 FINDIT  LEAU    -1,U      Move cursor pos. back...
0FF7 5A                 02320         DECB
0FF8 26   FB            02330         BNE     FINDIT     till start of match
0FFA A6   C4            02340         LDA     ,U        Save for non-destructive...
0FFC A7   8C 47         02350         STA     <FFLAG,PCR   cursor routine
                        02360
                        02370 *  Now wait for keypress
0FFF 86   AF            02390 CURSOR  LDA     #$AF      blue graphics block...
1001 A7   C4            02400         STA     ,U        to screen
1003 8D   21            02410         BSR     CRSKEY    try for keypress
1005 26   09            02420         BNE     PRESS     if keypress detected
                        02430
1007 A6   8C 3C         02440         LDA     <FFLAG,PCR  Else get first letter
100A A7   C4            02450         STA     ,U        and store on screen
100C 8D   18            02460         BSR     CRSKEY    and try for keypress
100E 27   EF            02470         BEQ     CURSOR    Keep trying if nothing
                        02480
                        02490
                        02500 *** GOT A KEYPRESS ***
                        02510
1010 E6   8C 33         02520 PRESS   LDB     <FFLAG,PCR  Restore first letter...
1013 E7   C4            02530         STB     ,U         to screen
                        02540
1015 81   0D            02550         CMPA    #$0D      Keypress was <Enter>?
1017 1027 FF72          02560         LBEQ    NXTLIN    If yes, do next lines
101B 81   45            02570         CMPA    #'E       Keypress was "EDIT?"
101D 27   16            02580         BEQ     EDIT      if yes
101F 81   03            02590         CMPA    #3        Keypress was <Break>?
1021 26   DC            02600         BNE     CURSOR    if not, keep trying for
                        02610 *                           legal key
                        02620
1023 7E   AE09          02630         JMP     $AE09     Exit here on <Break>
                        02640
                        02650
1026 108E 028A          02660 CRSKEY  LDY     #650      Delay value
                        02670
102A AD   9F A000       02680 DELAY   JSR     [$A000] Poll keyboard
102E 26   04            02690         BNE     KEYEND    Exit if keypress
1030 31   3F            02700         LEAY    -1,Y      else delay a little
1032 26   F6            02710         BNE     DELAY
1034 39                 02720 KEYEND  RTS               Return with or without keypress
                        02730
                        02740
                        02750 *** SET UP EDIT ROUTINE ***
                        02760
1035 BD   B95C          02770 EDIT    JSR     $B95C     Do an <Enter>
1038 9E   66            02780         LDX     <$66      Point to start of line
103A EC   02            02790         LDD     2,X       grab line number...
103C DD   2B            02800         STD     <$2B       and save for printout
103E 0F   D8            02810         CLR     <$D8      List flag: don't re-list
1040 EC   8C 04         02820         LDD     <BUFEND,PCR
1043 7E   8548          02830         JMP     $8548     Go Edit
                        02840
1046                    02850 FFLAG   RMB     1         <>0 if match found,
                        02860 *                           also holds letter under cursor
1047                    02870 BUFEND  RMB     2         End of line in print buffer
1049                    02880 LENTS   RMB     1         Length of target string
104A                    02890 TARGET  RMB     10        Buffer for Target string
          1054          02900 ZEND    EQU     .         End of program is here
                        02910
          0F00          02920         END     START     Exec address is at "START"
00000 TOTAL ERRORS
```

BUFEND	1047	DONE	0F82	FINDIT	0FF5	LENTS	1049	OK	0F66	START	0F00
CONT	0F9E	EDIT	1035	FOUND	0FE2	LOOP1	0F77	PRESS	1010	TABLE	0F2A
CRSKEY	1026	FFLAG	1046	JUMP	0F57	MATCH1	0FC5	PRINT	0FB8	TARGET	104A
CURSOR	0FFF	FIND	0F5F	KEYEND	1034	MATCH2	0FCD	PUTHI	0F1E	TEST	0FEA
DELAY	102A	FIND2	0F86	LENGTH	00F9	NXTLIN	0F8D	SHOWIT	0FF2	TITLE	0F5B
										ZEND	1054

Nexus presents
New Keys to Creativity!

CHROMA-SKETCH
The Picture Program Writer

CHROMA-SKETCH has an option that **writes a BASIC program** as you draw, paint and letter on the hi-res screen in **66 colors and shadings** (22 in each of 3 color-sets). Save program to disk or tape to recreate your picture from BASIC at any time, or save the screen as a binary image.

Dual cursors simplify arcs, circles, lines and boxes. Drawing aids include options for **full-screen crosshair cursor** and **"graph-paper" grid**. Automatic **preview** and **undo** commands available in 64K. **Help** key gives command and status display. If the program writer is enabled, you can return to **any previous stage** of your picture!

Draw complex shapes and CHROMA-SKETCH will redraw, rotate, reduce or enlarge them at any position in any color or pattern. Save the shape definitions to disk or tape for use in BASIC programs. **Paint in any dot pattern** including checks, stripes (horizontal or vertical) or others you define yourself. Draw dotted or twisted lines. Overlay color patterns for translucent effects.

CHROMA-SKETCH can be used with any combination of joystick devices including **Touch-Pad** or **Color Mouse.** If you prefer, you may use the fast, auto-repeat cursor keys to move diagonally, horizontally or vertically.

The fast, compact graphic programs you can write using CHROMA-SKETCH are **yours** to alter or use in any way you wish. These BASIC programs may be used for games, graphic adventures, educational software or on-screen slide shows.

Requires 32K Extended BASIC. Supports disk or cassette systems, all ROM versions, all graphic modes.

CHROMA-SKETCH

$29.95 ..... cassette
$34.95 ..... disk

FREE SHIPPING & HANDLING

The WIZARD Font
For Telewriter-64*

"I recommend Wizard for Telewriter devotees who want to do something nice for their hard-working word processor."
Scott Norman
Hot Coco, Feb. '85

"...I can attest to the relief that innovation offers the eyes after long periods of key-boarding." "Wizard is another CoCo winner..."
Charles Springer
Rainbow, Dec. '84

The pleasantly proportioned characters of the WIZARD font have **true descenders** and **optional end-of-line markers**. If you do not prefer this clear, attractive font, take advantage of our money back guarantee.

The simple modification of Telewriter-64 done by WIZ causes no change in the operation of the word processor or the amount of buffer space. You may choose to permanently install the font in a backup copy of Telewriter-64 or to quickly install the font each time you load an unmodified copy of Telewriter-64.

The end-of-line markers are useful for locating run-on spaces at the end of lines and between lines, and for counting spaces between paragraphs. They are essential for the complex on-screen formatting that tables or poetry often involve, as well as for keeping track of carriage returns in program code.

WIZ will run in any CoCo, **16-64K, disk or cassette.** The WIZARD font does not alter the characters produced by your printer.

Details of how the new font is added to Telewriter-64 may be requested when you order WIZ. Previous purchasers may request this information if they include their WIZ ID# and a SASE.

WIZ

$16.95 ..... cassette
$21.95 ..... disk

FREE SHIPPING & HANDLING

MONEY BACK GUARANTEE. If not completely satisfied, you may return the material within 30 days for a prompt refund. Specify the program and media desired. Programs supplied on disk and cassette are identical in all respects. Send check or money order. Personal checks — no delay. Next day shipping in most cases.

NEXUS
2601 Bridalwood Dr. No. 4
Knoxville, TN 37917

for additional information: (615) 522-1977. CIS 71575,1706
Write for a catalog. Dealer inquiries are welcome.
TN residents add 7% sales tax.

*Telewriter-64 is a trademark of Cognitec

****
*** NEWS FROM UNDER COLOR **** NEWS FOR UNDER THE COLORS ***
********** Compiled December 24, 1984 **********
** Send news to: Under Color, Box 6809, Roxbury, VT 05669 **
********** CompuServe 70136,1257 **********
****

UNDER COLOR now confirms that the new Korean Color Computer uses a PIGGYBACK 64K UPGRADE ... according to a reliable source, Tandy saves over $10 on the 16K machine, which is where Tandy believes most purchasers will STOP, since Tandy figures to spend $5 more for every 64K upgrade on the new machines. Does Tandy believe that the Christmas KoKos will GET TOSSED IN THE CLOSET after a while? Watch for LOWER COST third-party upgrades soon ... it looks like SPECTRUM PROJECTS is leading the race to market a KoKo 64K upgrade.

* * * * *

By the way, QUANTITY COST of bulk 64K memories is WAY, WAY DOWN. A check of mid-December prices shows BRAND NEW 64K memories from OKI costing only $2.50 each in 500 quantities ... that's $20 a set. WANNA MAKE A BUNDLE? Used, tested 64K chips can be obtained for as little as $1.25 in 400 quantity. Get together 50 orders and PAY ONLY $10 FOR A COMPLETE 64K UPGRADE. Remember 16K memories at $60 apiece? The price slide is happening to 64K's ... now that the 256K MEMORIES are being shipped at under $25 each.

* * * * *

And SPEAKING OF 256K, that's the maximum memory size of the almost-for-certain Color Computer '09. The so-called '09, at first victim of fits and starts, may be in the Radio Shack stores BEFORE MID-YEAR 1985. It is almost sure to have a 6809E processor, 128K MEMORY STANDARD expandable to 256K, 600 by 400 HIGH-RESOLUTION COLOR GRAPHICS, a 3 1/4" rigid floppy drive, direct-memory access along with LEVEL 2 OS-9 (supporting MULTI-USER capabilities), and a professional look. The price tag? WITHOUT A MONITOR, the rumor has the '09 coming in at $799. Is this a CHEVROLET COCO vs. a MERCEDES MACINTOSH?

* * * * *

What's the best $50 you can spend for software? A signup to CompuServe ... just to download MIKEYTERM, probably the finest Color Computer SMART TERMINAL PROGRAM you can't buy. Because IT'S FREE. Through the good will and generosity of programmer Mike "Guido" Ward, Mikeyterm has been released to the public domain. Initial users report that Mikeyterm IS AS GOOD AS ANY $250 TERMINAL PROGRAM ON THE MARKET. If you already use CompuServe, GO PCS-126 (The Color SIG) and get your hands on Mikeyterm.

* * * * *

Is there a 16/32-bit, 68000-BASED COMPUTER in your future? Dennis Kitsz is fielding any and all IDEAS FROM COCO USERS in search of the ultimately malleable personal computer. Dennis says NO BELLS AND WHISTLES, just a basic, wide-open design with lots of capability for expansion, plus 1/2 MEGABYTE OF MEMORY. Write: Dennis Kitsz, PROJECT 68000, Roxbury, Vermont 05669.

* * * * *

PCjr WINS HANDS DOWN, according to word from a well-known CoCo author. Last week he GAVE UP A COCO-PCjr COMPARISON STUDY because midway through, he began to feel that the COCO COULDN'T MEASURE UP, dollar for dollar, against the PCjr's new 1985 price breaks. Why did Dr. X can the project? "I didn't want to depress the CoCo faithful," he reported. The PCjr overtakes the CoCo as soon as you start to add peripherals, said Dr. X. UNDER COLOR is trying to convince him to RELEASE HIS FINDINGS anyway; readers wishing to have the report published please contact UNDER COLOR.

* * * * *

Please make the following additions and changes to your reference list from issue #1:

26-3005	Reserved (32K non-Extended Color Computer?)
26-3021	4-pin DIN to 5-pin DIN cable
26-3127	64K Extended Basic Color Computer 2, RF out
26-3128	64K Extended Basic Color Computer 2, NTSC out
26-3134A	16K "Korean" Basic (two 16Kx4 RAMs)
26-3136A	16K "Korean" Extended (two 16Kx4 RAMs)

* * * * *

COLOR MICRO JOURNAL, are you there? Two authors out looking for work called UNDER COLOR to say that CMJ was on the rocks. But as of this date NO CONFIRMATION of the demise of the newspaper-style CoCo publication was available. If CMJ goes, it will be the THIRD COCO MAG TO BITE THE DUST in recent times, along with Color Computer News and The Color Computer Magazine. Rumors of the death of Hot CoCo were FIRMLY SQUASHED once again by publisher Jeff DeTray. All the while, Lonnie Falk at The Rainbow relaxes in his easy chair, watching the unbecoming scuffle outside his office full of Southern hospitality. An update on CMJ will be provided in the next issue.

* * * * *

HAPPY NEW YEAR $7C1 from all of us at UNDER COLOR.

* * * * *

PRODUCT
64K Disk Utilities
Spectrum Projects
P.O. 21272
Woodhaven, NY 11412
64K
$21.95 disk

By Mark Haverstock

Spectrum Project's 64K Disk Utilities consists of three separate utility programs for the 64K Disk Color Computer: a 40K memory program, software print spooler, and ROMpak to disk converter. It's a fine example of a program that will let you see some of the capabilities of a 64K machine.

"40K," the first of the programs, lets you obtain 8K more RAM for Basic or assembly programming. This is accomplished by relocating the Extended Basic ROM from $8000 to $D800. Once 40K is executed, PRINT MEM displays 31015. Only 31015? Remember that 2K of disk space and four pages of memory for graphics are still reserved. Interestingly enough, the PCLEAR1 command will release those graphic pages, providing even more memory, but will register as −29913. (Note that numbers above 32767 are displayed as negative numbers.) Beside freeing more memory, it lets you modify Basic ROMs, which now reside in RAM. Two notes of caution apply to this program: Reset will wipe out any program in memory. Also, "40K" and "SPOOL64" may not be used at the same time. Any Color Computer user can now access some of that extra memory in a 64K machine without purchasing exotic DOS systems like OS-9.

"ROMpak to Disk Converter" is potentially the most appealing program in the package. Disk users can use most of their old ROMpaks without having to plug and unplug the disk controller by relocating the ROMpak programs to disk.

First the ROM is saved to cassette. This procedure involves covering pin 7 of the ROMpak to prevent it from auto-starting. The cartridge is then inserted into the machine and CSAVEM is executed. Afterwards, ROMcrack is run and the tape is read and relocated to disk. Once on disk, programs may be executed as any other assembly language program. Before trying this procedure, run ROMcrack. An excellent graphics display shows where pin 7 is located to prevent confusion.

Note that all ROMpaks will not relocate to disk using ROMcrack. Some, such as "Megabug" and "Stellar Lifeline," have routines which do not relocate well in different memory locations. Others, such as "Galactic Attack" and "Math Bingo," transfer to disk without problems.

"Software Disk Spooler" lets the upper 32K of memory act as a RAM buffer to feed your printer. If you've ever waited for a printer to finish several pages of text, you'll appreciate what the spooler can do. With SPOOL64 the computer is free for other tasks while the printer buzzes merrily on its way. Upon running the program, a main menu allows selection of six different bauds, from a slow 300 to a speedy 9600 baud.

Documentation for the three programs was minimal, consisting of two typewritten pages. "ROMcrack" and "Spooler" need clearer and more specific directions, as some are confusing. An example from ROMcrack: "Next, turn off your computer and remove a cartridge while the computer is on!"

Aside from the documentation, "64K Disk Utilities" is well-written and performs as advertised. It would be a useful addition for most Color Computer software libraries. *(end)*

Review: Super Deluxe Keyboard
By Jeffrey S. Parker

Super Deluxe Keyboard
Radio Shack Cat. No. 26-3016
Installation: $25.00
Keyboard: $39.95

Available: Radio Shack nationwide, or
Computer Plus
450 King St.
Littleton, MA 01460
617-486-3193
Price: $44.95

Caution! The catalog numbers for the low-profile keyboard and the new super deluxe keyboard are identical! Be sure to specify new full size keyboard when ordering!

The last bastion in keyboards has finally been breached by Radio Shack! After all those years, and all those keyboards, we have finally achieved a full-sized keyboard. The Super Deluxe keyboard is standard equipment on the new 64K Color Computer II. It is a full size, full travel keyboard, with the letters printed on the upper left corner of the keys, similar to the IBM format. The colors match the case, with the cursor, shift, clear and return keys in pure white. The Break key is still red.

I thought the ultimate had been reached when the low profile board came out, but one brief session at the Super Deluxe keyboard reveals the difference. The keys are high quality, with a good spring tension, and sensitive enough to register a light touch. The key tension is especially ideal for high speed touch-typing.

The Super Deluxe keyboard is sold only as stock equipment on the 64K Color Computer II at present, but will be available as an upgrade through Radio Shack dealers. Unfortunately, Radio Shack must do the installation, and they charge accordingly.

Reviews

There are several third party vendors from whom the board can be purchased, if one is willing and able to perform the installation. Computer Plus, in Littleton, MA, is one place that has them in stock. They are retailing the keyboard for $44.95, plus tax, and of course, shipping and handling.

If you have an F board or later version, the installation is extremely easy (remember to save your current keyboard if there's nothing wrong with it). If you have owned your computer for less than 90 days, opening it will void the warranty, so either wait for Radio Shack, or wait for 90 days. According to the New England Region supplier, Radio Shack centers will be carrying the new keyboards in a matter of a few weeks. Among other things, the Super Deluxe keyboard could make an excellent Christmas present! *(end)*

PRODUCT
Flexnet BBS
Oklahoma City, OK
(405)722-6809

By Bob Rosen

The Flexnet BBS is an unusual bulletin board system; it more resembles an operating system than a traditional BBS system. It is an adaptation of the Flex operating system from Technical Systems Consultants. The experienced user of Flex can go right to work on this BBS as though he had been on it for awhile; those unfamiliar with Flex may need a little time to get used to the system, but think of the opportunity to learn how this great operating system works!

I was introduced to Flex through my purchase of a TRS-80 Color Computer and was a little taken aback with the price of Flex programs. Most Color owners selected their computer for one reason: they didn't have a lot of money to spend. Well, you get what you pay for—especially with the limited operating system in the Color Computer.

The Flexnet BBS is a great service to Color owners in two ways. First, it allows the potential purchaser of one of the fine Flex implementations to try a Flex-like system before buying a copy. Second, there are lots of Flex utilities available on Flexnet, written by other Flexnet users. This is a great way to enhance your Flex library and learn a heck of a lot at minimum expense. The programs are in both source and object form. There are communications programs you can download in source form and assemble that will let you download machine code files. You would have to purchase a copy of Flex and an assembler if you wanted to use some of the Flex utilities, but there are also a lot of free download programs on the system that will run under RS Disk Basic. Another nice feature is that there are many knowledgeable people on the board who know a lot about Flex and the 6809 in general. If you pose a technical question on the BBS, there is a good chance it will be answered.

When you check into the board, there is a short greeting message and then the prompt FLEXNET. I thought a lot of being able to check into a BBS without having to give my name; this is the first I've ever seen. When you get the prompt, it's just like +++ in Flex; you give it a Flex command, and it's off and running. An example would be CAT; this would give you a catalog of the disk files. DIR gave me a more expanded directory of what was on the disk, and DIR 1 gave me a further directory of what was probably on another disk drive.

An interesting directory-type program was called FILES CMD; it was very handy because you can get a directory of files that have specific characteristics. For instance, if you only want a directory of files starting with "A," type FILES A; or if you want files with just the text, just type FILES TXT. To list a file that might be of interest, one would type LIST FILENAME EXT. The command LISTNEW [MM DD YY] will list all the files dated on or after the date you specify. This will let you look at all the files you want to without typing in each file name.

If you check into Flexnet and forget what to do, you can just type HELP and the system will walk you through it. The same goes for any command you may have a question about. For example, if you want more information about the command file SEND CMD, just type HELP SEND and the system will give you instructions on using the system's file transfer capability.

You can leave a program, message, or any other text-type file on the board using the Flex command BUILD FILENAME. Again, if you are not familiar with Flex, just type HELP BUILD.

If you want to know more, check into this very interesting BBS system. There are some pretty dull and poorly run BBS's in the country, but this is not one of them. My hat's off to Roger Walton and Bill Holland. *(end)*

continued from page 6

620 The derby takes full shape through the Paint command.

630–650 A series of characteristic spitcurls is formed across the forehead.

660–720 A series of sounds is assembled into F$, which when played at 720 proves to be the *Cuckoo Song*.

The program *Another Fine Mess* is achieved through highest resolution, PMODE 4,1, and it is monochromatic. By contrast the program *Winky* uses a few other tricks to achieve fairly high resolution along with a few colors and a bit of animation.

In *Winky* you can isolate effects by stopping the program after specific lines. For example, put in this line: 155 GOTO 155. This freezes the display to show you the effect of Line 150. Delete Line 155 and put in a new line: 165 GOTO 165. This will show the effect of Line 160. In this way you can move slowly through the program studying the effects of any given line.

I tend to use the Circle command to create the faces of characters—and somewhat symmetrical faces at that. Remember, I said I'm no artist, though I'm pleased with the results so far. *(end)*

I'm planning to build the 64K upgrade that appeared in the October, 1983 issue of The Color Computer Magazine, *for my revision D board computer. In the article it states that the ROM must be changed because it isn't a 1.1. However, trying* EXEC 41175 *indicates I have a 1.1 ROM. Do I still have to change ROMs, and what does this number indicate?*

Mark Patrick
W. Carrollton, OH

No, you don't need to change ROMs. The 1.1 ROMs correct several minor bugs present in the original 1.0 ROM and allow the computer to address more than just 16K of RAM.

I have the old grey Color Computer with 64K upgrade, HJL keyboard, and Green Mountain Micro Lowerkit II. I'm having space trouble with the internal modem. With the Lowerkit installed there isn't enough room for the two micro-mini switches to clear the RF shield. A schematic of the Modem I would be most helpful, but if that's not available could you label all terminals of S-1 (modem) and the installed micro-mini switches?

C.S. Holloway
Georgetown, CA

Your letter is very confusing! I can't find anything about an internally installed modem for the Color Computer. And the Modem I from Radio Shack (26-1175) is an external modem that you connect to the Color Computer via the serial port. (Please, when requesting help, include all the information you have on the hardware you're writing about. If I don't know what the hardware is, I can't help.)

In any case, I suggest you contact Green Mountain Micro and see if they have any suggestions to make the device fit around their Lowerkit II.

I purchased Radio Shack's replacement keyboard for my D revision Color Computer. I received a keyboard and a handful of parts, no instructions. My friendly dealer said Radio Shack would be glad to install the keyboard, but because I have heavily modified my unit, I would prefer not to do that.

Can you supply me with instructions? I have electronics experience and equipment.

Gerald Stewart
Rockbridge, OH

You should call National Parts and request the installation instructions for the replacement keyboard. In fact, the technical manual/installation instructions for any Radio Shack computer product can be ordered from National Parts. To get the correct part number for the manual you need, just add the letters MS in front of the hardware's part number. Thus, the part number for the new keyboard is MS260-3016. If you're still in doubt, call and ask the National Parts operator if you have the right number for what you want.

Radio Shack's National Parts address is: 900 East Northside Drive, Fort Worth, Texas, 76102, 817-870-5662.

Manufacturer's Announcements and New Products

Special Announcement Free Program

Memscan (Basic) is handy especially for programmers involved with graphics. Memscan is a memory pager utility that lets the user examine RAM in all display modes, locate and examine other programs stored in memory, examine memory allocation, and compare graphic pictures in various PMODES. 64K users can examine ROMs in the all RAM mode. Features jump to page or address, info on current display, manual and auto movement through memory, select display mode, return to previous location/display mode, and more. Cost: Free (not freeware)—send either $1 S&H or SASE with cassette and 40 cents postage, to NEXUS, 2601 Bridalwood Dr. #4, Knoxville, TN 37917 (CIS 71575,1706).

New Product Announcement Wiz

Wiz installs in Telewriter-64 a new on-screen character set (the Wizard font) designed for clarity and attractiveness. The font provides true descenders and optional end-of-line markers, and is vaguely reminiscent of calligraphy in style. It maximizes the readability of all display modes. The Wizard font may be installed each time you load Telewriter-64, or it may be permanently installed in a back-up copy of the word processor. Full instructions are included. The modification alters neither the size or operation of Telewriter-64. Wiz works in any Color Computer, 16K–64K, cassette and disk, and is supplied on cassette for $16.95 (S&H included). Contact Nexus, 2601 Bridalwood Dr. #4, Knoxville, TN 37917 (CIS 71575,1706).

New Product Announcement Screen Dump

SCPRNT is an assembly language screen dump for PMODE4 with a Basic driver that adds print lines to the left and right of the dumped image. This lets the image be labeled and marked externally, positioned horizontally, or simply allows text to be printed on the same page level as the image. The program also allows the dump to be halted at any of 27 levels on the screen. It is excellent for graphics. The program allows selection of a printed image 1, 2 or 3 times standard width. The user may select a white on black or black on white dump. It works with Radio Shack dot matrix printers and other comparable 7-bit graphics dot matrix printers. SCPRNT works on any Color Computer, 16K–64K, cassette and disk. It is supplied on cassette for $7.00 (S&H included). Contact Nexus, 2601 Bridalwood Dr. #4, Knoxville, TN 37917 (CIS 71575,1706).

Price Changes Occur On A Daily Basis. Please Call 1-800-343-8841

PRICE BREAKTHROUGH

Price Changes Occur On A Daily Basis. Please Call 1-800-343-8841

MEGADISK™ HARD DISK DRIVE SYSTEMS — **TOLL FREE ORDERING 1-800-343-8841**

For the **IBM/PC**, Tandy 1000, **TRS/80 Models I/III/IV/4P**, Compaq, Eagle, Sanyo, Tava, PC Workalikes, **Color Computers**, Apple/Franklin, Heath/Zenith, Max/80

Complete with Hardware, Cables, Software and Quikfit Installation

5 Megabytes Internal Mount	starting at	$ 449.95
10 Megabytes Internal Mount	starting at	599.95
20 Megabytes Internal Mount	starting at	1,199.95
5 Megabytes External System	starting at	649.95
10 Megabytes External System	starting at	849.95
20 Megabytes External System	starting at	1,299.95
3.3 Megabytes Kodak Backup System	starting at	599.95
Streaming Tape Backup System	starting at	899.95

DOS Systems Available: IBM/Heath — DOS, 1.0, 2.0, 2.1, 3.0, or later — Apple Franklin — DOS 3.3, Prodos TRS/80-LDOS, TRSDOS 6.x, Newdos/80, Dosplus, CP/M, COCO DOS, Max/80 LDOS, OS9

WARRANTIED FOR ONE FULL YEAR — PARTS AND LABOR — 24 HOUR SERVICE — CALL TOLL FREE — 1-800-343-8841

FLOPPY DISK DRIVES

Our Disk Drives are UL approved — Our Floppy Drive Cabinets and Power Supplies are Underwriters Laboratory Listed and have passed the required Federal Communications Part 15 Section B-EMI/RFI tests.

Warranty on all disk drives is one full years parts and labor. Warranty on floppy disk drive power supplies is five (5) years. In warranty or out of warranty service is 24 hour turn-a-round on all disk drives.

Tandon — Full Height Drives		
100-1	Single Sided 40 tk Bare	$128.00
	In Case with Power Supply	169.95
	Dual Drives in One Cabinet	329.95
100-2	Dual Sided 40 tk Bare	165.00
	In Case with Power Supply	209.95
	Dual Drives in One Cabinet	384.95
	Half High Drives	
65-1	Single Sided 40 tk Bare	125.00
	In Case with Power Supply	159.95
	Dual Drives in One Cabinet	309.95
65-2	**Dual Sided 40 tk Bare**	**145.00**
	In Case with Power Supply	**179.95**
	Dual Drives in One Cabinet	**349.95**
65-4	Dual Sided 80 tk Bare	165.00
	In Case with Power Supply	199.95
	Dual Drives in One Cabinet	359.95
TEAC Half High Drives		
55A	Single Sided 40 tk Bare	140.00
	In Case with Power Supply	169.95
	Dual Drives in One Cabinet	319.95
55B	Dual Sided 40 tk Bare	160.00
	In Case with Power Supply	199.95
	Dual Drives in One Cabinet	359.95
55F	Dual Sided 80 tk Bare	180.00
	In Case with Power Supply	219.95
	Dual Drives in One Cabinet	389.95
Apple/Franklin Disk Drives		
	35/40 Track in Case with Cable and Software	159.95
	Controller Card for Two Disk Drives	49.95
	Combination Price for Disk Drive and Controller	199.95
Commodore Disk Drives		239.95

COLOR COMPUTER DISK DRIVE SYSTEMS AND ADD IN PRODUCTS

40 Track Single Head Drive with Case, Power Supply, Cable Controller, Instruction Booklet, Diskettes	$279.95
Above with Dual Drives in One Cabinet	413.95
40 Track Dual Head with Case, Power Supply, Cable, Controller, Instruction Booklet, Diskettes	**289.95**
Above with Dual Drives in One Cabinet	**429.95**
128 Memory Upgrade Kit	129.95
Dual DOS Switch	19.95
With Second DOS System — JDOS, RSDOS, Micro DOS	48.00

TRS/80 HARDWARE

Model I Starter System — Delivered by UPS One Single Sided Disk Drive, Case, Cable, Power Supply, TRSDOS 1.3 and Manual	**$220.00**
Model III/IV Easy to Install Disk Drive Systems	309.95
Memory Upgrades — 4116 and 4164	starting at 1.50 ea.

ALL IN-STOCK ITEMS SHIPPED WITHIN 24 HOURS. SAME DAY SHIPPING PROVIDED BY REQUEST WITHOUT ANY EXTRA HANDLING CHARGES.

COMPUTERS

All of our computers come with a full one year Warranty — Parts and Labor — 24 Hour Service on all in stock parts.

IBM/PC-256K Two Floppy Drives, One Year Warranty	$1,895.00
With Monitor Card and High Resolution Monitor	2,195.00
IBM/PC-256K 5 Megabyte Drive, One Floppy Drive	2,495.00
With Monitor Card and Monochrome Monitor	2,695.00
10 Megabyte System	2,695.00
With Monitor Card and Monochrome Monitor	2,995.00
IBM Lookalike Monitor 800h x 650v resolution w/cable	199.00
TRS/80 Model 1V Computer 128K Dual Drive RS232	1,095.00

PRINTERS

Dot Matrix		
Star Micronics		
Gemini X-Series Parallel 120 CPS	starting at	$259.95
Delta 10/15 160 CPS	starting at	424.95
Radix 10/15 200 CPS	starting at	649.95
Panasonic 1090		249.95
Daisy Wheel		
Silver Reed 440 80 Column 12 CPS		315.95
550 132 Column 19 CPS		439.95
770 132 Column 36 CPS		895.00
Olympia 132 Column 14 CPS with Form and Tractor		399.95
Star Micronics 100 Column 18 CPS		352.95
Apple/Franklin Printer Interface w/Graphics and Cable		84.95
Printer Cables	starting at	19.95
Printer Paper — Microperf Edge 1000 Sheets		16.95

ELECTRICAL

Surge Protectors — Line Filters — SL Waber — 6 Outlets with Switch	$ 39.95
Uninterruptable Power Supplies — Panamax 250 Watt-15 Minute	399.95

MODEMS

Volksmodem 300 Baud	$ 59.95
Signalman Mark X Audodial	123.95
Mark XII 1200/300 Baud Autodial	284.95

ADD IN BOARDS FOR THE IBM

STB — Rio Plus 128K	$299.95
Super Rio 128K	219.95
Graphix Plus	283.00
Graphics Plus II	355.00
Parallel Printer Port	74.95
Serial Port	84.95
Monochrome Board with Parallel Printer Port	210.00
Quadram — Quadboard 512 128K	269.95
Quadcolor I	215.95
Quadcolor II	439.95

MISCELLANEOUS

Diskettes in 10 Pack	**from**	**$ 12.95**
Twoprint Switches	from	99.95
Disk Drive Cables	from	16.00
Maintenance Cleaning Kits		12.00
Parallel Printer Buffers 16K		149.95
Floppy Disk Drive Cables		
1 Drive		16.00
2 Drives		18.95
Heath/Zenith 2 Drive Cables — Shielded		39.95

SOFTWARE SUPPORT, INC.

1 Edgell Road, Framingham, MA 01701 (617) 872-9090 Telex-383425

Hours: Mon. thru Fri. 9:30 am to 5:30 pm (E.S.T.) Sat. 10 am to 3:30 pm

SERVICE POLICY — Our Professional Technical Staff Is Available To Assist You Monday Through Saturday.
WARRANTIES — Disk Drives — One Full Year Parts And Labor. Floppy Disk Drive Power Supplies — Five (5) Years.
SERVICE — 24 Hour Turn-A-Round On All In-Stock Parts. Price Changes Occur On A Daily Basis. Please Call

Toll Free 1-800-343-8841

For Our Latest Price Saving Specials.

DYNACALC®

CoCo's Best & Fastest Spreadsheet System

ACCLAIMED BY THE EXPERTS

"DYNACALC is my choice for a CoCo spreadsheet."
Dan Downard, RAINBOW, September, 1984.

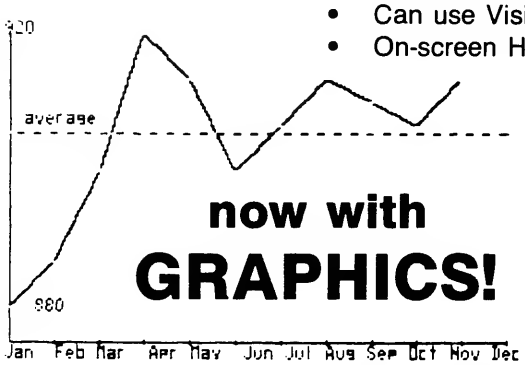
"Eat your heart out, Lotus 1-2-3!"
Scott Norman, HOT CoCo, October, 1984.

NOW ONLY $99.95

Built-in Features:

- 51 x 24 Display with Lower Case
- Super-fast Smart Screen Refresh
- Auto-Repeat Keyboard Driver
- Keysaver (TM) repeats last command x times
- Disk Operating System (works just like ROM DOS)
- Easy communication with BASIC/DOS programs
- Two-way communications with PRO-COLOR-FILE ★Enhanced★
- Outputs to ASCII Word Processors like Telewriter-64
- Fast 16-Digit Arithmetic with Scientific Functions
- Summation, Mean, and Standard Deviation Functions
- Logical Functions with String & Numeric Comparison
- String locate command to navigate large worksheets
- Sort full or partial worksheet by columns or rows
- Line, Bar, Hi/Lo/Close, Circle Graphs
- Full Graphics captioning and overlay facility
- Graphics Drivers for all popular Printers
- Joystick/Mouse Driver for Cursor Movement
- Works with any ROM versions, even JDOS
- 33k Available Worksheet Space
- Up to 256 Columns or 256 Rows
- Can use VisiCalc worksheets & training materials
- On-screen Help Messages

RAINBOW CERTIFICATION SEAL



now with GRAPHICS!

FOR 64K DISK SYSTEMS
available from

COMPUTER SYSTEMS CENTER
42 Four Seasons Center #122
Chesterfield, MO 63017
(314) 576-5020
10 to 6 Mon.-Fri.

NEW ADDRESS

CSC

or your local DYNACALC dealer

Price US$99.95
Outside North America add $10 postage

CANADA- CDN $129.95
RGS MICRO INC.
759, VICTORIA SQUARE 405
MONTREAL H2Y 2J3
TEL.: (514) 287-1563
ORDER LINE ONLY ★ ★ ★
QUEBEC-ONTARIO-MARITIMES
800-361-5338
WESTERN CANADA 800-361-5155

MasterCard
VISA